Contribution to Enhance IPSec Security by a Safe and Efficient Internet Key Exchange Protocol

Ahmim Marwa, Babes Malika, Ghoualmi Nacira
Badji Mokhtar University
Annaba, Algeria / Networks and Systems laboratory
Algeria
ahmim.marwa@gmail.com, {malikababes, Ghoualmi}@yahoo.fr

ABSTACT: *IPSec is a suite of protocols that provides security for internet communications at the IP layer. The security properties of IPSec mainly depend on the key exchange protocols where the efficiency and security of the key management are important parts of IPSec. Internet Key Exchange (IKE) protocol is the most common mechanism for the two hosts to exchange key materials. However, IKE is complex and vulnerable due to attacks such as (DOS, Replay and Man in the middle). In this paper, we propose a new IKE protocol based on D-H. This protocol uses three round-trips the exchange message. The advantages of our contribution are: one phase (vs. two phases on standard IKE), Best efficiency ie. optimizes transmission time (vs. longer negotiation time). The security analysis and formal verification using Automated Validation of Internet Security Protocols and Applications (AVISPA) show that our contribution can resist to various attack types such as (Replay, DOS, man in the middle). We compare our IKE with other IKE protocols; the proposed protocol is more secure with less computation complexity.*

Keywords: Internet Protocol Security (IPSec), Security Association (SA), Internet Key Exchange Pprotocol (IKE), Security Analysis, Attacks

Received: 10 May 2013, Revised 27 June 2013, Accepted 30 June 2013

1. Introduction

Nowadays, the Internet connects millions of people around the world and allows for immediate communication and access to a seemingly limitless amount of information [1]. However, this medium has its risks such as loss of privacy, loss of data integrity, identifies spoofing and denials of service are some of the major threats on the Internet.

Internet Protocol Security (IPSEC) deals with some of these problems by providing security services such as data source data integrity, authentication, confidentiality, access control, data privacy and protection against replay attack [2].

To provide the security services for IPSec, the first step is to establish mutual authentication between entities at the beginning of the connection and the negotiation of session keys and confidential parameters to be used during the connection [2].

IKE is used to provide the first step of IPSec. In this paper, we propose a new IKE protocol to enhance the security of IPSec.

Our contribution is organized as follows. In section II, we present a description of the protocols IPSec and IKE. We give, in section III the related works. We outline in section IV the proposed protocol. Section V describes the security analysis. A formal verification with AVISPA tools of the proposed protocol is given in section VI. An evaluation of the performance between our IKE protocol and other is shown in section VII. Section VIII concludes the paper.

2. IPsec and IKE Overview

2.1 IPsec

The IPSec architecture is a set of protocols, algorithms designed by the Network Working Group of IETF to provide security services such as the authentication, access control, confidentiality (encryption) and data integrity at the IP layer.

In IPSec, before providing the security services, it must establish mutual authentication between peers unknown to each other and shared session keys. Then it negotiates and exchanges the parameters for the connection. These parameters include: IP initiator address, IP destination address, security parameter index (SPI), security protocol identifier (SPId), IPSec protocol mode, sequence number counter, lifetime, encryption algorithm and key materials. The above parameters allow creating security association (SA) [3].

IKE is used by IPSec to establish SA dynamically, automatic negotiation of parameters (key, encryption algorithm...) and authentication. The establishment and negotiations of the security association IKE is formed by itself but the security association IPSec is formed by IKE [3, 4].

IPSec security services are provided by two extension headers, the Authentication Header (AH) [5] and the Encapsulating Security Payload (ESP) [6], and through the use of cryptographic key management procedures and protocols (IKE).

• **The Authentication Header [5]:** The protocol is used when both integrity and authenticity of IP package or its load capacity must be protected, but not necessarily the confidentiality of the packet itself.

• **The Encapsulating Security Payload [6]:** The protocol is used to encrypt and encapsulate either the transport layer payload or the entire IP packet.

• **IKE:** IKE is explained in the following section.

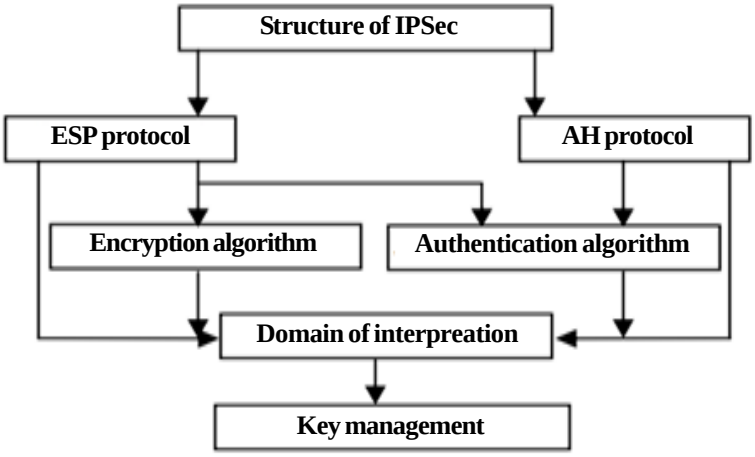

Figure 1. Structure of IPSec [4]

2.2 IKE

The IKE protocol structure is defined by the Network Working Group of IETF to set up a security association of IPSec [7]. IKE is used to provide the security association for IPSec. Firstly, doing mutual authentication between two IPSec peers. Then, it

establishes a shared secret key. Finally, it negotiates parameters IPSec SA [3]. The security Association (SA) is a data structure which is used to store and protect all the confidential parameters (security policy and key) between one device and another one. The IKE consists of two phases: Phase 1 for establishing IKE SA and Phase 2 for establishing IPSec SA. Since Phase 2 negotiation is protected by IKE SA, the negotiation of Phase 1 is of major concern.

There are eight variants of Phase 1 of IKE. That's why there are two modes of exchange (Main and Aggressive), each mode has four different authentication methods (public key signature, pre-shared, public key encryption, and revised public key encryption) figure 2 depicts the IKE Main mode using public key signature for authentication, which is the basic form of the eight variants [8, 3].

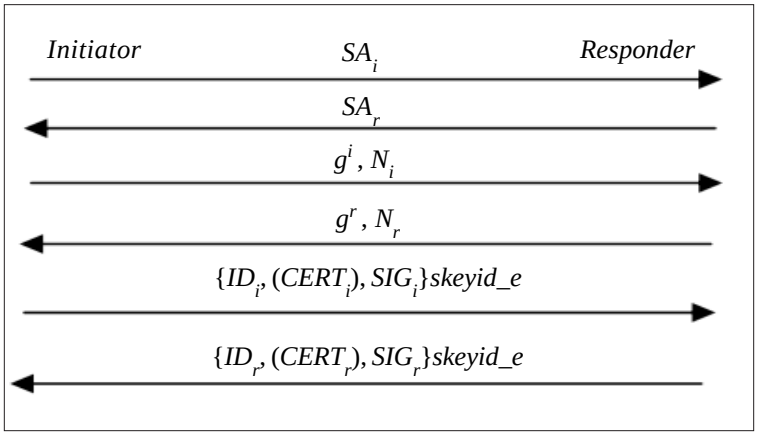

Figure 2. IKE Main mode with signature authentication [3]

"*In Figure* 2, *g, N, ID, CERT and SIG are Diffie-Hellman exponentials, nonce, identity, public key certificate and signature, respectively. Subscripts/superscripts i and r are used to represent the data generated by or belonging to initiator or responder, respectively. Data like CERT put in parenthesis can be omitted if not necessary.* SA_i *is the IKE SA proposal proposed by the initiator, and* SA_R *is the IKE SA reply from the responder. After exchanging D-H public keys and nonce values* (*in the second round-trip*), *the communication parties begin to authenticate each other using the third round-trip exchange. The confidential information including ID, CERT, and SIG would be encrypted by the symmetric key skeyid_e, which is basically derived from the two nonce values and the* g^{ir} *shared session key.* SIG_i (SIG_r) *is the digital signature of applying the initiator's* (*responder's*) *private key to sign a hash value*" [3]. A detailed description of IKE protocol is given in reference [7, 8].

3. Related Works

In the operation mode of IPSec, the first step is to establish the security association where we use the IKE protocol. The first version of IKE is based on Diffie-Hellman that is vulnerable to active attacks such as: Denial of Service (DoS), Man in the Middle and the replay attack. Several studies have criticized the vulnerability of the IKE; In order to correct it, several improvements have been proposed. In this section, we detail the most significant works in this context.

In [9], Zhou investigates some flaws and weaknesses in the IKE protocol specified in RFC 2409 such as flaw authentication in phase 1 and he examined the failure of identity protection in the main mode protocol with digital signature for authentication. He proposed some changes to remedy these weaknesses.

P.C .Cheng [8] presents the detailed design of IKE protocol and its performance evaluation. The description and analysis of original IKE for the IPSec are presented in [10].

A comparative study in two criteria (Security and performance) of Successor protocols IKE such as (IKEv2, JFK, and SIGMA) were done by Haddad and Mirmohamadi in [1]. Ningning Lu and all have overcome weakness of the safety versions IKE protocol [11].

A new IKE protocol is proposed in [2], this protocol is resistant to DOS attack and the CPU exhaustion attack. A new Internet

Key Exchange protocol is suggested in [3], which is inspired by Haddad's protocol [2].

In [4] the authors analyze the security of IPSec, introduce a dynamic pre-shared key generation method to improve the security of IPSec. A new IKE protocol is proposed in [12], a hash function of the public encryption key and signature key are used to generate the secret session key, instead of using nonce and a cookie.

Cas Cremers [13] provides a formal analysis of IKE protocol, which allows to find several previously unreported weaknesses on the authentication properties of IKE.

We give a general overview of the architecture and possible attacks of IKE protocols in table 1.

Ref	Security metric						
	M1	M2	M3	M4	Efficiency		M5
					NBM	NBP	
[8] [9]	/	/	/	/	6 in ph 1 4 in ph 2	2	/
[2]	yes	yes	/	yes	3 in ph 1 4 in ph 2	2	yes
[3]	yes	yes	/	yes	4	1	yes
[4]	/	/	/	/	6 in ph 1 4 in ph 2	2	yes
[12]	yes	yes	/	/	6 in ph 1 4 in ph 2	2	/
[1]	yes	yes	/	yes	4	1	yes

Table 1. A synthesis of IKE protocols studied

Notes: M1 - PFS; M2-Known key security; M3- Resilience to Replay attack; M4- DOS defense; M5- Resilience to man in the middle attack; NBM- Number of messages; NBP- Number of phases.

This synthesis has shown IKE protocol is complex and vulnerable to various attack types (DOS, Man in the middle and Replay).

4. Our Proposed IKE Protocol

Our work aims at building a secure IKE protocol with less computation complexity.

In this work, we use the D-H key exchange protocol [14] to build a new IKE protocol. The latter is composed of six messages. The first four messages are used to establish IKE-SA and the two behinds messages under protection by shared session key are used to establish IPSec-SA. Unlike related work, our protocol can resist to various attack types such as (DOS, man in the middle and replay) and several security properties are verified.

4.1 Notations used

- ID_a : Identity of initiator *A*;
- ID_b : Identity of initiator *B*;
- SA_i : A list of cryptographic proposals of the initiator (security association proposals of IKE);
- SA_r : Cryptographic protocols selected by the responder from the list sent by the initiator (security association selected of IKE);
- P_A : Password of intiator;

- P_B : Password of responder;
- $\oplus$: XOR;
- $||$: Concatenation;
- $N1$: Random number;
- $SA_{ipsec\ 1}$: A list of cryptographic proposals of the initiator (security association proposals of IPSec);
- $SA_{ipsec\ 2}$: Cryptographic protocols selected by the responder from the list sent by the initiator (security association selected of IPsec);
- $H(.)$: Hash function;
- K_{AB} : The derived session key by two-party;
- $E_{K_{AB}}(.)$ Encryption using a symmetric cryptosystem with key K_{AB};
- AS : Authentication server.

4.2 Protocol description

The proposed Internet key exchange protocol between Initiator (Alice) and responder (Bob) is depicted in Figure 3. It consists of 6 steps:

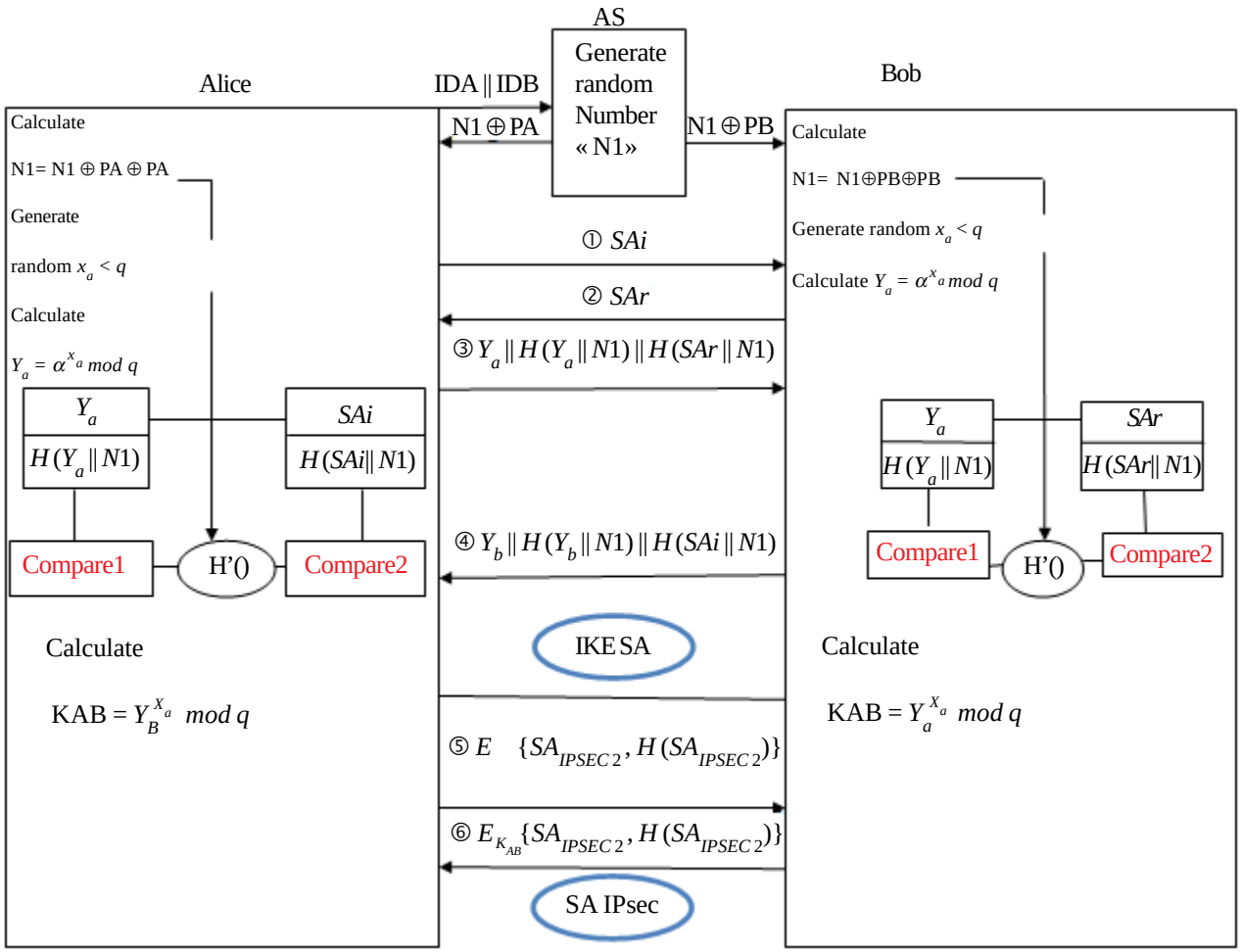

Figure 3. Our IKE protocol

Step 1: *Initiator* $\rightarrow$ *Responder* : SA_i

The Initiation sends to the responder a series of cryptographic proposals for SA_IKE (Security Association IKE).

Step 2: *Responder* $\rightarrow$ *initiator* : SA_r
The responder selects SA_r from SA_i according to its preference and sends SA_r to the initiator. If the responder does not agree for a SA, it can reject the entire list of SA and sends back an error in the second message.

Step 3: *Initiator* $\rightarrow$ *Responder* : $Y_a \,||\, H(Y_a \,||\, N1) \,||\, H(SAr \,||\, N1)$
Upon receiving the responder message, initiator performs the following operations:

- Selects a random number X_a;
- Computes: $Y_a = \alpha^{x_a} \bmod q$;
- The initiator (Alice) sends a request message to the AS that includes the IDA and IDB;
- The AS: Upon receiving initiator message (IDa || IDb), sends $N1 \oplus PA$ to the initiator and $N1 \oplus PB$ to the responder;
- The initiator calculates $N1 = N1 \oplus PA \oplus PA$ and sends $Y_a \,||\, H(Y_a \,||\, N1) \,||\, H(SAr \,||\, N1)$ to the responder.

Step 4: *Responder* $\rightarrow$ *initiator*: $Y_b \,||\, H(Y_b \,||\, N1) \,||\, H(SAi \,||\, N1)$
Upon receiving initiator message, responder performs the following operations:

- Calculates: $N1 = N1 \oplus PB \oplus PB$;
- Calculates $H'(Y_a \,||\, N1)$ by Y_a from the initiator and $N1$ from AS and $H'(SAr \,||\, N1)$ by SAr (association security selected by the responder) and $N1$ from AS;
- Verifies whether $H'(Y_a \,||\, N1) \stackrel{?}{=} H(Y_a \,||\, N1)$ and $H'(SAr \,||\, N1) \stackrel{?}{=} H'(SAr \,||\, N1)$. If the verification fails, responder terminates the execution; otherwise, the responder Selects a random number X_b, Computes $Y_b = \alpha^{x_b} \bmod q$ and sends $Y_b \,||\, H(Y_b \,||\, N1) \,||\, H(SAi \,||\, N1)$ to initiator.

Step 5: *Initiator* $\rightarrow$ *Responder* : $E_{K_{AB}}\{SA_{IPSEC\,1}, H(SA_{IPSEC\,1})\}$
The initiator *performs* the following operations:

- Calculates $H'(Y_b \,||\, N1)$ by Y_b from the responder and $N1$ from AS and $H'(SAi \,||\, N1)$ by SAi (association security proposals by the initiator) and $N1$ from AS;
- Verifies whether $H'(Y_b \,||\, N1) = H(Y_b \,||\, N1)$and $H'(SAi \,||\, N1) \stackrel{?}{=} H'(SAi \,||\, N1)$If the verification fails, initiator terminates the execution; otherwise, the initiator calculates K_{AB} and encrypts the $SA_{IPSEC\,1}$, $H(SA_{IPSEC\,1})$ using the encryption key (K_{AB}) previously generated and sends it to the responder.

Step 6: *Responder* $\rightarrow$ *initiator*: $E_{K_{AB}}\{SA_{IPSEC\,2}, H(SA_{IPSEC\,2})\}$
Upon receiving initiator message, responder performs the following operations:

- Decrypts the received encrypted message using K_{AB};
- Selects a $SA_{ipsec\,2}$ from $SA_{ipsec\,1}$ according to its preference; if the responder does not agree for an SA then it can reject the entire list of $SA_{ipsec\,1}$ and sends back an error in the second message; otherwise, the responder sends $E_{K_{AB}}\{SA_{ipsec\,2}, H(SA_{ipsec\,2})\}$ to initiator.

5. Security Analysis

The security properties assured by our protocol:

- **Known-Key Security:** In our protocol, since K is calculated by two numbers random (x_a, x_b) so each complete negotiation should result in a unique shared session key. Therefore, the compromise of one shared session key should not compromise keys in other sessions.
- **Resilience to Replay attacks:** Our protocol can resist replay attack, random value $N1$ assures that the response is fresh and has not been replayed by an opponent.

• **DoS Defense:** Two types of flooding packets have to be considered: Msg3 and Msg5. According to our protocol, one forged Msg3 would totally cause responder to spend time for twice hashing, one XOR function. On the other hand, one forged Msg5 would totally cause the responder to do once hashing and one symmetric encryption. Since all of these operations are simple and could be done very fast, a DoS attack would not find it too easy to totally exhaust the responder's CPU time unless such attack is lasted for a fairly long period of time.

• **Efficiency:** Our proposed IKE needs only one phase, which consists of three round-trips exchange messages. The first four messages are used to establish IKE SA and the two behind, under protection messages with shared session key are used to establish IPSec SA.

• **Resilience to Man-in-Middle Attack:** By our protocol, $N1$ is the secret information only between initiator and responder, the use of $N1$ can be effective to authenticate the two parties and resilience to Man-in-Middle Attack.

• **Resilience Control Key:** Our protocol, since $K_{AB} = Y_B^{X_a} \bmod q$ no single entity is able to force the shared session key to be a pre-selected value.

6. Formal Analysis Using AVISPA & SPAN

We model the proposed protocol using AVISPA tool [15] to verify security properties assured by our protocol.

6.1 The AVISPA

The avispa is a push-button tool for the automatic validation of the protocols and applications Internet sensitive to the security. It provides a modular and expressive formal language to specify the protocols and their properties of security; it integrates different back-end which implements a variety of automatic techniques analysis of machine [15]. The AVISPA uses the intruding model Dolev-Yao where any intruder (adversary active or passive) can spy on any transmitted message, the mount masquerading (impersonation attacks) and replay attacks (to modify or inject any message) but to follow the perfect cryptography, i.e. the intruder cannot break cryptography [16]. The AVISPA framework has been shown in Figure 4 below.

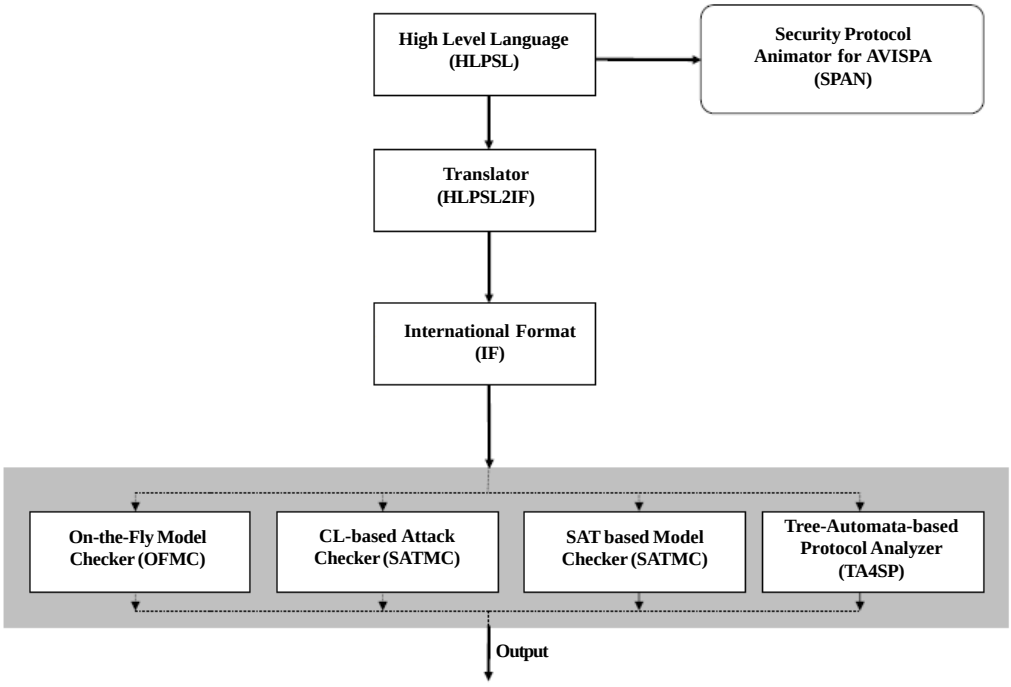

Figure 4. Pictorial diagram of the AVISPA

6.2 Protocol Specification

The proposed protocol was modeled in a formal language called HLPSL and written in the file with extension hlpsl (IKE.hlpsl).

This language is based on roles. There are three basic roles "*Alice*", "*Bob*" and "*AS* ". In this stage, we only present one of the basic roles bob shown in Figure 5 as an example. After defining the basic roles, we have to define the 5 composed roles describing the sessions of the protocol. Next, a top-level role is defined. This role contains global constants and the initial knowledge of intruder and composition of more sessions. Finally, the goal section is used to specify the security objectives, i.e. the properties that enable AVISPA tools to search the attacks. The witness and request events are for authentication property whereas the Secrecy events are used to check the shared secrecy among the agents "*alice*" and "*Bob*".

```
role bob (…) played_by B def=
local const   init  State := 0 transition
 State = 0 /\ Rcv(SA1') =|>
      State':= 1 /\ SA2':=new()
                 /\Snd(SA2)
 State = 1  /\ Rcv(xor(N1'.pb)) =|>
State':= 2
  /\ witness(B,S,sk2,N1')
   /\ N2':= xor(xor(N1',pb),pb) …….
 State = 4 /\ Rcv({SA3'.H(SA3')}_KAB)
=|> State':= 5   /\ Snd({SA3.H(SA3)}_KAB)
            /\ secret(KAB,sec_KA,{A,B})
end role
```

Figure 5. Role Bob

The validation of the modeled protocol representation was conducted by using a tool called SPAN. After that, this protocol was executed against the modeled intruder to verify desired security goals to verify the strengths and weakness by using AVISPA tools (OFMC).

6.3 Analysis of results

We choose the back-end OFMC of the AVISPA framework to verify the security by our protocol:

• **Man in the Middle Attack Check:** The description of the environment role is given below can detect "*the man in the middle attacks*" if it exists.

```
role environment() def=
intruder_knowledge = {a,b,s}
  composition
    session
(a,b,s,g,snd,rcv) /\ session
(a,i,s,g,snd,rcv) /\ session
(i,b,s,g,snd,rcv)
end role
```

Figure 6. Role of Environment 1

The Results (figure 7) with this description have reported the protocol is safe against attacks "*man in the middle*".

• **Resilience to Replay attack**: When you use the-sessco option, OFMC initially will carry out a research with a passive intruder to check if the honest agents can carry out the protocol, then to give the intruder the knowledge of some "*normal*" meetings between the honest agents. The results show that our protocol can resist against replay attack.

• **Delov-Yao Model Check:** At the end, the depth we have chosen for the search is eleven and the output of model checking results are shown in Figure 7. As shown in the figure, there are totally 1624 nodes have been searched in 2.37s. From these

results, we can conclude that the proposed protocol can reach the design properties and it is secure under the test of AVISPA using the OFMC back-end with a bounded number of sessions.

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  C:\SPAN\testsuite\results\IKE.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 2.37s
  visitedNodes: 1624 nodes
  depth: 11 plies
```

Figure 7. Results reported by the OFMC back-end

7. Peformance Evaluation

7.1 Complexity Analysis

To evaluate the complexity of the proposed IKE protocol, we focus on the following operations: pseudo-random, hash function, secret key en/decryption, public key en/ decryption number message in phase I, number message in phase II, modular computation (subtraction, addition, exponential) and the XOR operation. We give a complexity analysis between our proposed protocol and other version IKE protocol in table 2.

	Pseudo Radom	Hash function	N. of msg to create SA-IKE	N. of msg to create SA-IPSec	Secret Key en/ decryption.	Modular computation (exp, add, sub)	XoR	Public key en/ decrption
IKEv1 A/B	3/2	1/1	6	4	4/3	2/2 (exp) 0/0 (add) 0/0 (sub)	0/0	1/1
[2] A/B	0/0	2/2	2	2	2/4	2/2 (add) 5/6 (exp) 0/0 (sub)	0/0	0/0
IKEv2 A/B	1/1	1/1	2	2	2/2	2/2 (exp) 0/0 (add) 0/0 (sub)	0/0	1/1
Our IKE protocol A/B	0/0	2/2	4	2	1/1	0/0 (add) 0/0 (sub) 2/2 (exp)	2/2	1/1

Table 2. Performance Comparaison

Our IKE protocol uses: four messages for mutual authentication, the establishment of a shared key and the creation IKE-SA; two

messages to create IPSec-SA. In addition, it use once the symmetric encryption; two Hash function. Thus, our schema can provide a higher-level security with less computation complexity.

8. Conclusion

In this paper, we proposed a new Internet Key Exchange protocol based on *D-H*. It uses three round-trips exchange message. The first four messages are used to establish IKE-SA and the two behinds messages under protection by shared session key are used to establish IPSec-SA. Therefore, there are several advantages which make our protocol better than other protocols. Between these advantages we find: one phase (vs. two phases on standard IKE), best efficiency ie. optimizes transmission time (vs. longer negotiation time). The security analysis and formal verification using Automated Validation of Internet Security Protocols and Applications (AVISPA) show that our contribution can resist to various attack types such as (DoS, man in the middle and replay).

Reference

[1] Haddad, H., Mirmohamadi, H. (2005). Comparative evaluation of successor protocols to Internet key exchange IKE). *In*: Proceedings of the IEEE Intl. Conf. on Industrial Informatics, August, p. 692-696.

[2] Haddad. H., Berenjkoub, M., Gazor. S. (2004). A Proposed Protocol for Internet Key Exchange (IKE), *Electrical and Computer Engineering, Canadian Conf*, May.

[3] Ming-Yang Su, Jia-Feng Chang. (2007). An efficient and secured internet key exchange protocol design. *In*: Proc. of the fifth annual conference on Communication Networks and Services Research (CNSR'07), p. 184-192.

[4] Liangbin Zheng, Yongbin Zhang. (2009). An Enhanced IPSec Security Strategy, *International Forum on Information Technology and Applications*, China, p. 499.

[5] Kent, S., Atkinson, R. (1998). IP Authentication Header (AH), RFC 2402, Noveber.

[6] Kent, S., Atkinson, R. (1998). IP Encapsulating Security Payload (ESP), RFC 2406, November.

[7] Harkins, D., Carrel, D. (1998). The Internet Key Exchange (lKE), RFC2409, Nov.

[8] Cheng, P. C. (2001). An Architechture for Internet Key Exchange Protocol. *IBM System Journal*, 40, 721-746.

[9] Zhou, J. (2000). Further analysis of the Internet key exchange protocol. *Computer Communications*, 23, 1606-1612.

[10] Radia Perlman, Charlie Kaufman. Analysis of the IPSec Key Exchange Standard, *Tenth IEEE International*, p.150 – 156.

[11] Ningning Lu, Huachun, Zhou Yajuan Qin. (2008). A Comparison Study of IKE Protocols 08 Proceedings of the International Conference on Mobile Technology, *Applications, and Systems*.

[12] Nagalakshmi, V., Rameshbabu, I., Avadhani, P. S. (2011). Modified protocols for internet key exchange (IKE) using public encryption key and signature keys. *In*: Proc. of the eighth international conference on Information Technology: New Generation, p. 376-381

[13] Cas Cremers. (2011). Key Exchange in IPsec revisited: Formal Analysis of IKEv1 and IKEv2 , *In*: Proceedings: 16th European Symposium on Research in Computer Security.

[14] Nan Li. (2010). Research on Diffie-Hellman Key Exchange Protocol, *In*: Proceedings: 2nd Computer Engineering and Technology (ICCET).

[15] Avispav1.1User Manual. (2006). Available at http://www.avispa-project.org.

[16] Atanu Basu, Indranil Sengupta, Jamuna Kanta Sing. (2012). Formal Security Verification of Secured ECC Based Signcryption Scheme, *The Second International Conference on Computer Science, Engineering & Applications* (ICCSEA)