ORIGINAL ARTICLE
Journal Section

ESIKE: An efficient and secure internet key exchange protocol

Marwa Ahmim[1] | Ahmed Ahmim[2] | Mohamed Amine Ferrag[3] | Nacira Ghoualmi-Zine[1] | Leandros Maglaras[4]

[1]Networks and Systems Laboratory (LRS), Department of Computer Science, Badji Mokhtar - Annaba University, Annaba, 23000, Algeria

[2]Mohamed-Cherif Messaadia University - Souk Ahras, Souk Ahras, 41000, Algeria

[3]Department of Computer Science, Guelma University, Guelma, 24000, Algeria

[4]School of Computer Science and Informatics, De Montfort University, Leiceste, UK

Correspondence
Marwa Ahmim, Networks and Systems Laboratory (LRS), Department of Computer Science, Badji Mokhtar - Annaba University, Annaba, 23000, Algeria
Email: ahmim.marwa@gmail.com

Funding information

The use of Internet key exchange protocols in IP Security architecture and in IoT environments has vulnerabilities against various malicious attacks and affects communication efficiency. To address these weaknesses, we propose a novel efficient and secure Internet key exchange protocol (ESIKE), which achieves a high level of security along with low computational cost and energy consumption. ESIKE achieves perfect forward secrecy, anonymity, known-key security, and untraceability properties. ESIKE can resist several attacks, such as, replay, DoS, eavesdropping, man-in-the-middle and modification. In addition, the formal security validation using AVISPA tools confirms the superiority of ESIKE in terms of security.

KEYWORDS
IKE, Formal analysis, Attacks, Computational optimisation.

1 | INTRODUCTION

Internet of Things connects the virtual world to the real world through smart objects. The increased use of intelligent objects in our daily life implies a higher need for security of transmitted data. Security, especially the key management process and the authentication of entities, represents a major problem for IoT devices due to limited resources of entities namely memory, computation and battery.

In order to solve these problems, there are several authentication schemes, and authentication and key management protocols (DTLS and IKE) are proposed in the literature.

Unfortunately, the majority of proposed authentication works start with an initialization phase where the transmission channel is supposed secure and they share parameters which will be used during the authentication phase as secret values. In [1] the authors propose an authentication scheme based on a secure channel, and they share important parameters in this channel. Thereafter, the shared parameters are used as secret values in the authentication phase. In addition, they introduce the concept of a valid authentication period for IoT. Khemissa et al. designed an authentication scheme based on HMAC operation, where they used a secure channel to share parameters used for the following communication phases [2]. Alshahrani et al. [3] presented an authentication protocol based on secure channel for exchanging an asymmetric key, where they introduced the concept of a counter that is used to verify authentication. In [4], the authors proved the vulnerability of the authentication schemes proposed by Kalra and Sood, and Chang et al. and proposed an improvement.

6LoWPANs created by the IETF. It aim is to define an adaptation layer placed between the data link layer and the network layer in order to ensure the transmission of IP datagrams over IEEE 802.15.4 links.This layer performs fragmentation, reassembly and compression processes. In IoT, the internal security of messages transmitted between nodes is performed by several protocols such as DTLS and IKE [5].

The DTLS is developed to protect the CoAP (Constrained Application Protocol) application layer [5]. A compression mechanism is required for the DTLS protocol.This mechanism can compromise end-to-end security dimensions. In addition, the key management and authentication scheme based on Elliptic Curve Cryptography does not meet the needs of the IoT environment because of the fragmentation process of large messages made during adaptation. So, the retransmission and reorganization of messages is necessary [5].

Internet Protocol version 6 is considered an ideal solution for the IoT. The IPv6 use Internet Security Protocol (IPSec) to secure the information exchanged. The IPSec is part of the IETF protocols suite that provide the security of Internet Protocol (IP). It is designed to protect the exchanges in IP networks. This protocol aims to ensure several security dimensions such as: data source authentication, confidentiality, data integrity and access control.

IPSec provide security services combining two protocols. The AH protocol [6] is used to guarantee both authenticity and integrity of IP packets. The ESP protocol [7] is used to ensure two security dimensions such as authentication and confidentiality. Before providing thse security dimensions, IPSec decides on the security parameters to be applied such as: the cryptographic algorithms, the security protocols (AH, ESP), the secret keys, and the choice between transport mode or tunnel mode.

The Security Association (SA) is used by the IPSec in order to manage the confidential parameters. It consists of all the necessary information to make processing IPSec on an IP packet [8] [9].

So, the IPSec needs a way to exchange security association information. It uses the IKE protocol to provide these capabilities. Figure 1 depicts the components of the IPSec.

The IKE protocol is the principal part of the IPSec implementation. It is used to negotiate the secret keys between the two parties, the initiator and responder. It is used to create security associations (SA) that define how the traffic between the two parties will be protected [10] [11] [12]. The main security requirements of the IPSec protocol depends to IKE protocol.

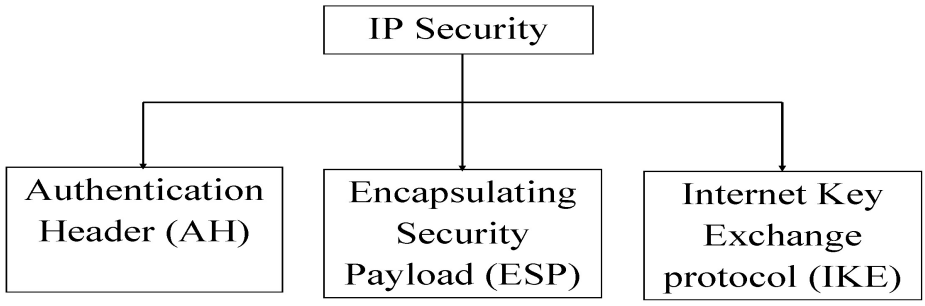

FIGURE 1 The components of IPSec.

In [13], the authors propose a protocol which is used to compress the headers of the IPSec protocol. Despite the compression, this protocol remains inapplicable in the IoT because it uses a heavy key management protocol.

In this article, we present the security analysis of the original IKE and his successor. Then, we propose a new efficient and secure internet key exchange protocol applicable in the IoT environment. Although the theoretical verification of security protocols is typically used to validate the security requirements, it remains inadequate. Consequently, AVISPA formal protocol analysis tools has been employed to verify the different security properties of ESIKE.

The rest of this article is organized as follows. In section 2, we present the original IKE protocol and his successor. Section 3, illustrates the proposed protocol. Then, the theoretical analysis and formal validation using AVISPA tools are discussed in section 4. In section 5, we present a performance comparison of ESIKE and other key management protocols existing in the literature. Finally, Conclusion is given in section 6.

2 | THE IKE PROTOCOLS

The Internet Key Exchange IKEv1 protocol is described in RFC 2409 and is used in IPSec. It is composed of two steps. The first step is used to establish an IKE SA and create secrets keys and the second step is used to establish IPSec SA [14].

Moreover, IKEv1 protocol has two modes of exchange during step 1 (Main Mode-MM and Aggressive Mode-AM), and one mode of exchanges during step 2. The difference between the two modes of the first step is that MM mode provides identity protection, and it is composed of six messages. In contrast, AM mode does not offer an identity protection, and it is composed of only three messages. In addition, both MM and AM support four authentication methods based on pre-shared key, public key signature, revised public key encryption and public key encryption. Figure 2 represents the process of IKE protocol [14].

The IKEv1 protocol has been analyzed by several researchers. The first formal analysis was performed by Meadows in 1999. He proved that the IKEv1 is vulnerable to DoS attacks [15]. In [16], Zhou showcased the weaknesses of IKEv1 protocol during step 1 and proposed some modifications in order to reduce these weaknesses. In 2001, Perlma and Kaufman have performed another analysis of IKEv1. Where, they demonstrated the weaknesses of IKEv1 when using a pre-shared key, and they give suggestions to improve and simplify it [17].

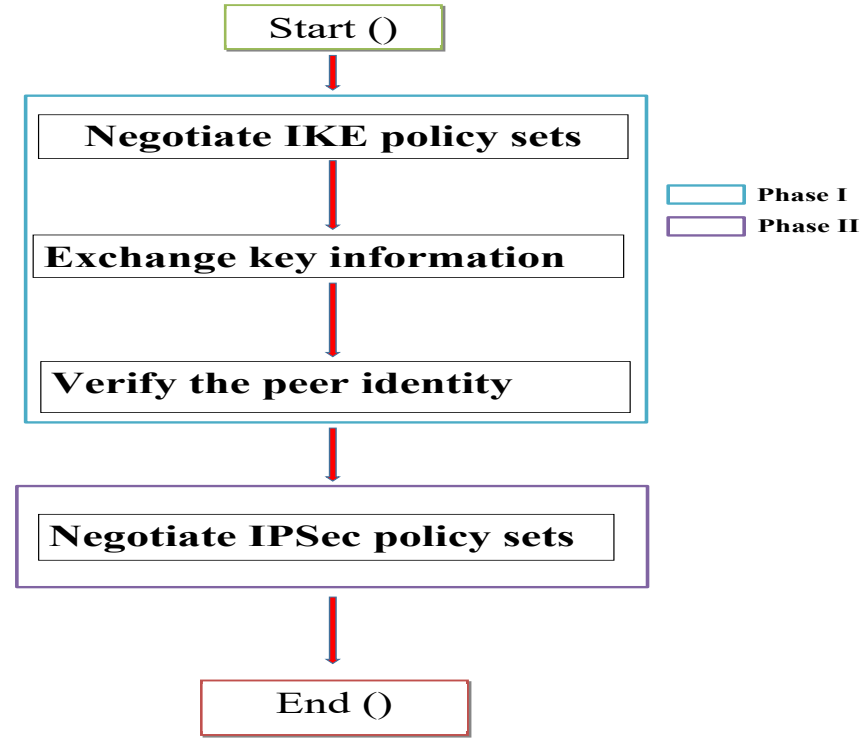

FIGURE 2 IKE process

To mitigate these weaknesses, several protocols have been proposed. Aiello et al. proposed a JKF protocol to remedy the vulnerability against DoS attacks [18]. Afterward, in [19] the authors propose a modified IKE protocol that withstands to DoS attack. In 2005, RFCs [20] propose a new IKE protocol appointed as IKEv2 . Then, in 2006, Smith et al. have found two succesfull DoS attacks in JFK protocol [21]. Later in 2007, Su and Chang [10] proposed an efficient version of Haddad et al. protocol [19].

The formal analysis of the IKEv2 performed by AVISPA project demonstrated the vulnerability of IKEv2 against the DoS attacks [22]. The work presented in [23] is an enhancement of the IKEv2 protocol [20]. Another IKEv2 protocol analysis was conducted in [24]. The authors of this work introduced some updates to improve the resilience of the IKEv2 protocol against DoS attacks. A novel IKE protocol was proposed in [25], in this proposal, the generation of secret session key depends on a hash function. The parameters of this function are the public encryption key and the signature key, used as an alternative of nonce and cookie. Most recently, the proposition made in [26] provided

an IKE protocol formal analysis. This analysis permits the discovering of different flaws on the IKE authentication properties which were not formerly reported. Taking advantage of the first version of IKE, authors in [27] propose a novel protocol protocol based on IKEv1. In [28], the authors focused on the resistance against cyber-attacks and proposed a new IKE protocol which they claim to be robust to several types of attack types such as man-in-the-middle, DoS and replay.

In [29] Lavanya and Natarajan propose a lightweight IKE protocol for IoT. This protocol has security weaknesses, mainly against a man-in-the-middle attack.

3 | THE PROPOSED IKE PROTOCOL

In this section, we explain our proposed protocol, entitled ESIKE, that reassures can be used as an Efficient and Secure Internet Key Exchange protocol. ESIKE is composed of two exchange messages. Using these messages, the sender and the receiver share their private key, establish IPSec-SA and authenticate each other.
In contrast to other similar works, the proposed protocol can satisfy all the security properties of the key management protocol and it can withstand to several attack types such as eavesdropping, man-in-the-middle, replay, modification and DoS. Moreover, ESIKE only uses a basic operations namely Exclusive OR providing protection against both passive and active attacks while reassuring at the same time low computational cost and limited energy consumption.

3.1 | Notations

We present in this subsection, the symbols used in the ESIKE (see Table 1).

3.2 | Protocol description

ESIKE is based on ECDH with modifications that guarantee the security properties and the efficiency of the key management protocol. As depicted in Figure 3. there are three steps in ESIKE:
Step 1: Initiator to Responder: SA_{ipsec1}, M_1, $Verf_1$, Y_i, T_s
The initiator generates two random numbers r_i , $w_i \in$ [1, n-1]. Then, it makes the following operations:

- It calculates: $y_1 = \mathrm{H}(r_i \parallel w_i)$, $K_1 = y_1.T_r$, $M_1 = y_1 \bigoplus K_1$, $Verf_1 = E_{K1}\ \{\ \mathrm{H}(SA_{ipsec1} \parallel y_1 \parallel ID_i \parallel T_s\ )\ \}$, $Y_i = ID_i \bigoplus y_1$.
- It sends SA_{ipsec1}, M_1, $Verf_1$,Y_i, T_s

Step 2: Responder to Initiator: SA_{ipsec2}, M_2, $Verf_2$, Y_r
When the responder receives the initiator message, it makes the following operations:

- It chooses a SA_{ipsec2} from SA_{ipsec1}
- It calculates: $K_1^{'}$, $y_1^{'}$, $ID_i^{'}$
- It decrypts the $Verf_1$
- It calculates: $Verf_1^{'} = \mathrm{H}(SA_{ipsec1} \parallel y_1^{'} \parallel ID_i^{'} \parallel T_s)$, then if the verification fails ($Verf_1^{''} \neq Verf_1$), it ends the execution.

Symbol	Definition
I	Initiator
R	Responder
ID_i	The identity of I
ID_r	The identity of R
P	ECC point generator
r_i, r_r	Secret keys of I and R
T_i, T_r	Public keys of I and R, $T_i = H(r_i \parallel w_i) . P$, $T_r = H(r_r \parallel w_r) . P$
T_s	Time stamp
w_i, w_r	Secret keys of I and R
H	Hash fonctions
SA_{ipsec1}	The security association proposals by I
SA_{ipsec2}	The security association selected by R
K_1	The shared session key between I and R
K_{ir}	The calculated session key between I and R
E_{K1}	Symmetric encryption with the secret key K_1

TABLE 1 Notations used in ESIKE

If the verification is successful, the responder confirms the identity of the initiator and makes the following operations:

- It calculates: $y_2 = H(r_r \parallel w_r)$, $M_2 = y_2 \bigoplus K_1$, $Verf_2 = E_{K1}\ \{ H(SA_{ipsec2} \parallel y_2 \parallel ID_r) \}$, $Y_r = ID_r \bigoplus y_2$.
- It sends SA_{ipsec2}, M_2, $Verf_2$,Y_r

Step3: When the initiator receives the responder message, it makes the following operations:

- It calculates: $y_2^{'}$, $ID_r^{'}$
- It decrypts the $Verf_2$
- It calculates: $Verf_2^{'} = H(SA_{ipsec2} \parallel y_2^{'} \parallel ID_r^{'})$, then if the verification fails ($Verf_2^{''} \neq Verf_2$), it ends the execution. Else, the initiator confirms the identity of responder.

4 | SECURITY ANALYSIS

In this section, we start by checking the security precepts and the resistance of our protocol against cyber-attacks. Then, we present the obtained results of the formal verification of our protocol specifications obtained from the OFMC tool.

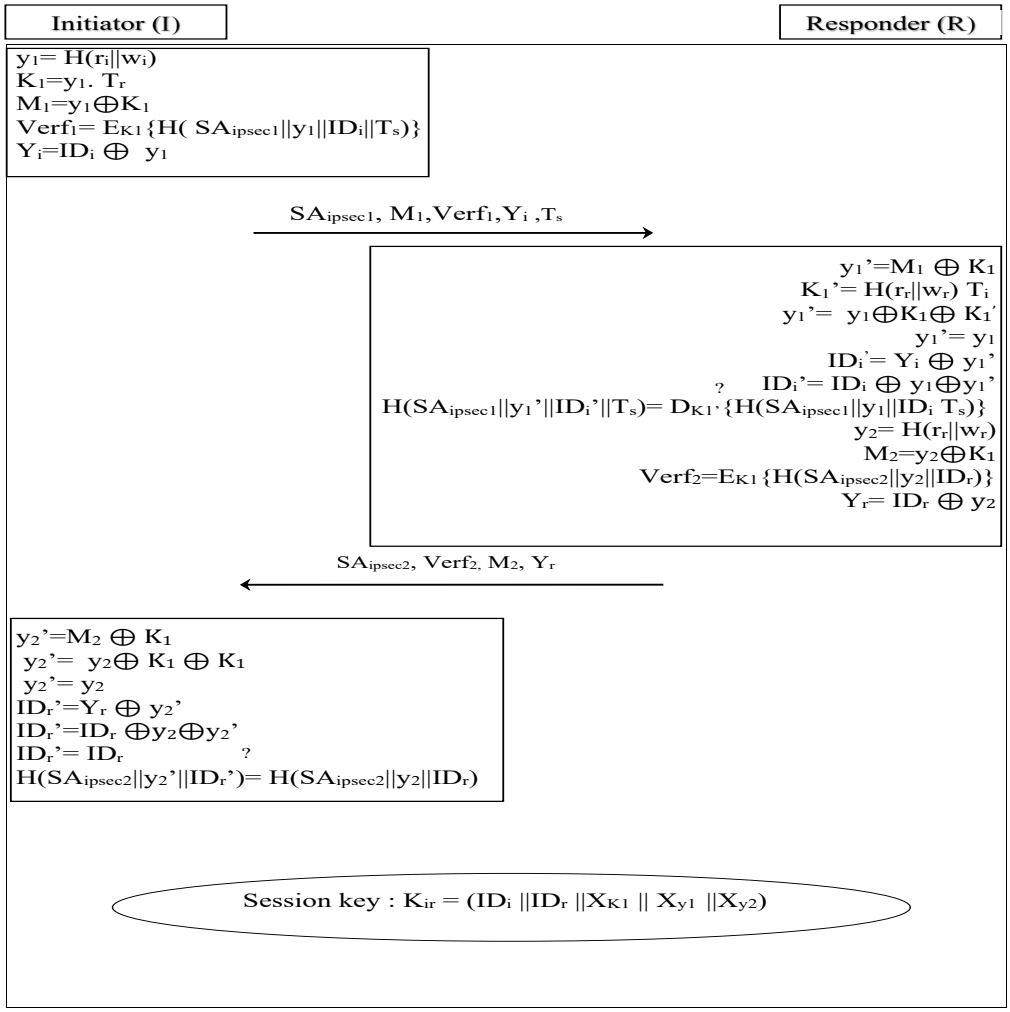

FIGURE 3 ESIKE.

4.1 | Theoretical Analysis

Precept 1. ESIKE preserves the perfect forward secrecy property, the discovery of the session key by an adversary. ESIKE doesn't allow the adversary to find any previous session keys.

Proof. Suppose that the adversary knows secret key K_{ir}, he attempts to determine the session key $K_{ir} = ID_i \parallel ID_r \parallel X_{K1} \parallel X_{y1} \parallel X_{y2}$ for past sessions. However, in order to derive the session key, the adversary needs to know the secret key K_1, the identity of initiator and the identity of responder. the secret key K_1 depend to the random values r_i, r_r, w_i and w_r. Solving K_1 to get r_i, r_r, w_i and w_r correspondents to the resolution of Elliptic Curve Discrete Logarithm Problem (ECDLP). Likewise, the secret key of ESIKE is protected by a hash function. So, ESIKE has perfect forward secrecy property.

Precept 2. ESIKE preserves the Known-key security, in case of compromise of previously generated session key does not help the adversary compromise other session keys.

Proof. Let us assume the adversary knows the session key derived by the ESIKE . The adversary is unable to generate the previous and future session keys, due to the fact that the generation of the session key requires the knowledge of K_1, X_{y1} and X_{y2}. To calculate K_1, X_{y1} and X_{y2}, the adversary needs to know r_i, r_r, w_i and w_r. Note that, computing of r_i, r_r, w_i and w_r from the K_1 is unfeasible, because it is equal to the resolution of ECDLP. Hence, known-key security is satisfied in our proposed protocol.

Precept 3. ESIKE provides the Key-Compromise Impersonation, if the long-term private key of the node is found; an adversary cannot impersonate as another node to communicate with the compromised node.

Proof. It is assumed that the long-term private key of the compromised node "I", r_i is found by the adversary "Eve". It is clear that the adversary "Eve" can impersonate "I". However, in order to deceive any other node "R" that is communicating with I, Eve needs the session key, $K_{ir} = ID_i \parallel ID_r \parallel X_{K1} \parallel X_{y1} \parallel X_{y2}$. Thus, the adversary "Eve" needs to have the X_{y1}, X_{y2} and the identity of "R". Solving X_{y1} and X_{y2} to get r_r, w_i and w_r is equal to solving of ECDLP. Therefore, ESIKE has this property.

Replay attack: ESIKE is robust to replay attacks. Suppose that an adversary intercepts old exchanged messages, and tries to replay them in order to impersonate another's node identity. The adversary cannot impersonate the sender or receiver node because new random numbers are generated for each authentication, and these with the use of timestamps detect the replay attack.

Efficiency: ESIKE is based on ECDH that uses a small key size. The use of shorter key length requires less space for key storage, therefore saves bandwidth for key transmission and reduces the arithmetic computation costs. These characteristics make elliptic curve cryptosystem the best choice to enhance security in IoT. ESIKE has only one-phase, which constitutes of two messages. These messages are used to share private keys, create security association of IPsec, and perform a mutual authentication between sender and receiver nodes. So, ESIKE is effective in IoT.

DoS robustness: In ESIKE, there is one type of flooding packets $Message1$. if $thefalsified - Message1$ is sent to the receiver node. This falsified Message causes the responder node to execute once en /decryption and 3 hash functions. All these operations can be performed quickly. So, a DoS attack cannot prevent the receiver node from operating normally.

Eavesdropping attack: ESIKE can withstand to the Eavesdropping attack. Let us assume an adversary "Eve" intercepts the message exchange between the sender and receiver nodes (SA_{ipsec1}, M_1, $Verf_1$, Y_i, T_s). The key K_{ir} cannot be compromised by the attacker because the construction of the K_{ir} includes ID_i, ID_r, and the hash of four values generated randomly.

Man-in-the-middle attack: Suppose that an adversary is spying on the communication channel between the sender and receiver nodes. The man-in-the-middle attack cannot succeed because the receiver node computes $Verf_c$. Then, it verifies $Verf_c$ with $Verf_{send}$. In these steps the adversary fails and ESIKE can withstand to man-in-the-middle attack.

Anonymity and untraceability properties: Suppose that Msg 1 and Msg 2 are intercepted by the attacker. During the authentication, we use nonce and timestamps, which ensure the freshness of the messages. In addition, the ID_i, ID_r, are sent in a hidden way.

Modification attack : ESIKE withstands to the modification attack. Suppose that the adversary “Eve” intercepts a message transmitted over a network and attempts to modify it, this won’t be possible. Because our protocol uses the parameters such as $Verf_1$,and $Verf_2$,which are used to check the integrity of the message.

4.2 | Formal analysis

The usefulness of AVISPA is that automatically validates the security of Internet protocols and applications that are sensitive to security. The language proposed by AVISPA is expressive, formal, and modular. Hence, it is used to describe the protocols to be evaluated and the security properties related to them. AVSPA incorporates several verification tools (back-ends); these analyzers allow the implementation of different automatic analysis techniques. In addition, AVISPA extends a standard for intruders called: Dolev-Yao intruder model. An intruder is considered an active or a passive adversary. It is assumed that an intruder is able to spy on any transmitted message; he can also pretend to be an authorized user and performs a masquerade or an impersonation attack. Moreover, he can update the content of any message or inject other messages in order to launch a replay attack. However, in order to follow the perfect cryptography, it is assumed that an intruder is incapable to break cryptography.

The AVISPA framework is depicted in the below Figure 4. The High Level Protocol Specification Language (HLPSL) is an AVISPA’s special language. It is primarily used to describe a protocol and its properties. After that, the protocol description is transparently translated by the HLPSL2IF module to an Intermediate Format (IF), which is a lower level language. The backends used by AVISPA are: On the fly Model Checker (OFMC), CL based Attack Searcher (CL-AtSe), SAT-based Model Checker (SATMC) and Tree Automata based Protocol Analyzer (TA4SP). These four back-ends utilize the IF presentation as an input to execute the protocol analysis. Thus the validation results are extracted as an output format that testifies whether the protocol is safe or not, in case of insecurity the flaw causes are also specified. [22],[30],[31].

To analyze ESIKE with the AVISPA tool, the following steps were performed:

Step1: the modelling of the protocol is done via the HLPSL formal language and saved in a hlpsl file. The basis of this language is roles. In ESIKE, we use Alice and Bob as the two essential roles. Alice represents the initiator and Bob represents the responder. Figure 5 illustrates the basic role of “Alice”.

Step2: the roles which describe the different sessions of the protocol are defined in this step. A top-level role is also defined. This role encloses global constants, the intruder initial knowledge, and the other sessions composition.

Step3: security objectives are the properties used to examine potential attacks on the protocol. These objectives are specified in the goal section. Two types of events are used in this step, the first one is used for authentication property which includes witness and request events. The second type includes secrecy events which are used to check the shared secrecy between the agents "Alice" and "Bob".

Step4: the representation of the modelled protocol is validated via the SPAN tool [30]. In order to confirm the security of ESIKE, the OFMC back-end executes this protocol against the modelled intruder. This step permits the verification of the desired security goals and the identification of the protocol’s strengths and weaknesses in term of security.

The ESIKE is verified in the OFMC back-end, and the result is shown in Figure 6. Consequently and according to

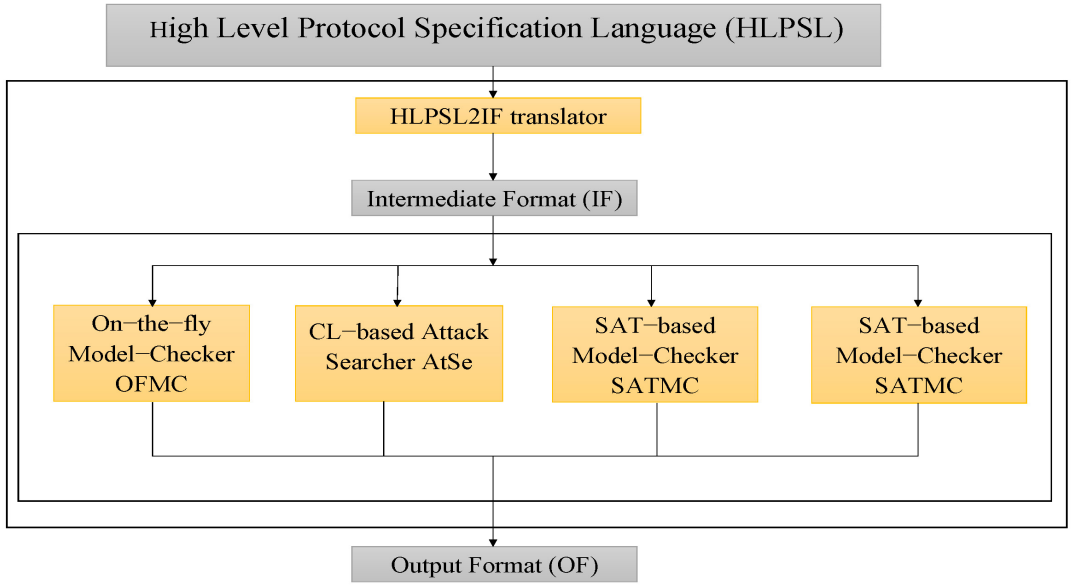

FIGURE 4 The framework of the AVISPA tool.

this result, ESIKE can resist to passive and active attacks.

5 | PERFORMANCE COMPARISON WITH COMPETITIVE PROTOCOLS

In this section, we give a performance and security comparison for ESIKE with authentication schemes in IoT and IKE protocols. Figure 7 illustrates a comparison between the messages number in phase 1 and phase 2 of ESIKE and five previous related protocols ([16];[16];[20]; [27];[29]).

ESIKE uses two messages to make mutual authentication between two communicating nodes, negotiate parameters of IPSec-SA and establish a shared secret. Our protocol uses the least number of messages.
The comparative study (see Table 2) of our protocol with IKEs and the authentication schemes for IoT existing in the literature shows that our protocol is better.
ESIKE does not use a secure channel in the initialization phase. It ensures the security requirements of key management protocol and resists against various attacks types using at the same time smaller key size, making ESIKE an efficient protocol for use in IoT environments.

Notes: Pseudo Radom function (C1); Hash function(C2); Private key encryption-decryption (C3); mathematical operations(C4): Exponential (Ept), Multiplication (Mlt); Public key encryption-decryption(C5). Perfect Forward Security (PF); Known Key Security (KS); Protection to Modification Attack (PM); Protection to Reflection Attack (PR); Resistance to Replay Attack (RR); Protection to DoS Attack (DS); Protection to Man in the Middle Attack (MM); ✓ means 'satisfy' and × 'not satisfy'.

```
role alice(P: text,
A,B : agent,
H: hash_func,
SND_B,RCV_B : channel(dy))
played_by A def=
local
State : nat,
F1: function,
Ta,Taa,Tbb,R1,R2,W1,IDa,IDb,Verf1,K1,K2,Verf2
,Y1,Sa1,Sa2,M1,M2,Yi,Y2,Yr,N,Z,Nonce,Nonce1:
text
const ok:message,
             sec_a_K1:protocol_id

init State := 0
transition
1.State=0/\RCV_B(start)=|>
            State':=1 ..................
2. State = 1 /\..................
                         /\ Yi':= xor(IDa', M1')
                         /\ Sa1':=new()
                         /\ Nonce':=new()
                         ..................

/\SND_B(Sa1',M1',Verf1',Yi',Nonce')
2. State = 3 /\
RCV_B(Sa2',M2',Verf2',Yr',Nonce1') =|>
   State':= 5
                   /\ Y2':=xor (M2',K1')
                   ..................
                   /\  request (A,B,k1,K1)
3. State =5 /\ ( Z' = Verf2')
           =|>..................

end role
```

FIGURE 5 Alice role.

6 | CONCLUSIONS

The IP-based Internet of Things is a technology that eliminates the boundaries between the physical world and the virtual world. The connected objects are generally heterogeneous and limited in resources. The exchange of the secrets between these objects and the authentication between them is a major problem.

Although many Internet key exchange protocols have been proposed recently, most of them suffer from many weaknesses like vulnerabilities to various attacks, high complexity of protocols structure, and low communication efficiency. In order to overcome these shortcomings, we propose a new IKE based on ECDH. The proposed solution, entitled ESIKE, is robust against several attacks (man in the middle, modification, DoS, eavesdropping and replay) and it offers

```
% 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: 0.35s
  visitedNodes: 208 nodes
  depth: 6 plies
```

FIGURE 6 The formal validation results of ESIKE by OFMC back-end.

Reference	*Security metric*							*Computation and efficient*				
	PF	*KK*	*PM*	*PR*	*RR*	*DS*	*MM*	*C1*	*C2*	*C3*	*C4*	*C5*
[16]	×	×	×	×	×	×	×	10/10	0/0	5/5	2/2(Ept)	0/0
[20]	✓	✓	×	×	×	×	✓	6/6	0/0	3/3	2/2(Ept)	0/0
[27]	✓	✓	×	×	✓	✓	✓	10/10	0/0	3/3	0/0(Ept)	0/0
[29]			×				×	0/0	2/2	1/1	2/2(Mlt)	1/1
[3]	✓	✓	×	×	✓	×	×	2/2	2/2	1/1	0/0(Ept)	0/0
[1]	✓	✓	×	×	✓	×	×	2/2	7/7	0/0	0/0(Ept)	0/0
[4]	✓	✓			✓	×	×	1/1	4/4	0/0	0/0(Ept)	0/0
[ESIKE]	✓	✓	✓	✓	✓	✓	✓	0/0	3/3	1/1	0/0(Ept)	0/0

TABLE 2 Comparison between ESIKE and previous related protocols

all the security features required by a key management protocol. ESIKE consists of the exchange of two messages. These messages are used to share private keys between entities, establish IPSec-SA and authenticate each other. Furthermore, ESIKE is efficient in terms of performance compared to existing key management and authentication protocols. Finally, the formal verification using AVISPA tools confirms the superiority of of ESIKE in terms of security.

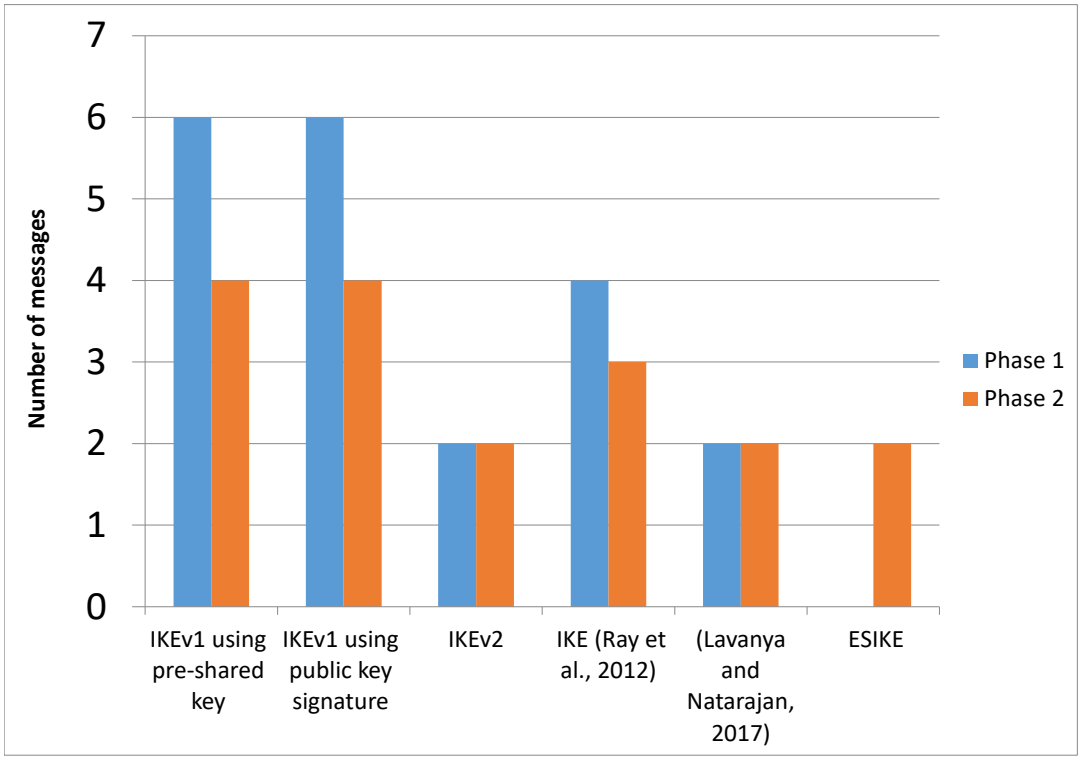

FIGURE 7 Messages number in phase 1 and phase 2 of IKE protocol.

7 | DECLARATIONS

- Funding: Not applicable
- Conflicts of interest: Not applicable
- Availability of data and material: Not applicable
- Code availability:the custom code availability in AVISPA tool

References

[1] Chuang YH, Lo NW, Yang CY, Tang SW. A lightweight continuous authentication protocol for the Internet of Things. Sensors, Vol 18, No 4, pp1-26 2018;.

[2] Khemissa H, Tandjaoui D. A Lightweight Authentication Scheme for E-Health Applications in the Context of Internet of Things. International Conference on Next Generation Mobile Applications, Services and Technologies, pp90-95 2015;.

[3] Alshahrani M, Traore I, Woungang I. Anonymous IoT Mutual Inter-Device Authentication Scheme Based on Incremental

Counter (AIMIA-IC). 7th International Conference on Future Internet of Things and Cloud (FiCloud),IEEE, pp31-41 2019;.

[4] Wang KH, Chen CM, Fang W, Wu TY. A secure authentication scheme for Internet of Things. Pervasive and Mobile Computing, Vol 42, pp15-26 2017;.

[5] Glissa G, Meddeb A. 6LowPSec: An end-to-end security protocol for 6LoWPAN. Ad Hoc Networks 2019;82:100–112.

[6] Kent S, Atkinson R. RFC 2402. IP Authentication Header (AH). URL: http://www faqs org/rfcs/rfc2402 html 1998;.

[7] Kent S, Atkinson R. RFC 2406: IP encapsulating security payload (ESP). IETF, November 1998;.

[8] Allard F, Bonnin JM. An application of the context transfer protocol: IPsec in a IPv6 mobility environment. Communication Networks and Distributed Systems, Vol1, No1, pp110–126 2008;.

[9] Thomas J, Elbirt AJ. Understanding Internet Protocol Security. Information Systems Security, Vol 13, No 4, pp39–43 2006;.

[10] Su M, Chang JF. An efficient and secured internet key exchange protocol design. In Proceedings of the fifth annual conference on Communication Networks and Services Research (CNSR'07), Fredericton, New Brunswick, Canada, pp184–192 2007;.

[11] Zheng L, Zhang Y. An Enhanced IPSec Security Strategy. In Proceedings of International Forum on Information Technology and Applications, China, pp499 – 50 2009;.

[12] Cheng. An Architecture for Internet Key Exchange Protocol. IBM System Journal, Vol 40, No 3, pp721–746 2001;.

[13] Raza S, Voigt T, Jutvik V. Lightweight IKEv2: a key management solution for both the compressed IPSec and the IEEE 802.15.4 Security. Proceedings of the IETF workshop on smart object security, Vol 23 2012;.

[14] Harkins D, carrel D. RFC2409: The Internet Key Exchange (IKE). URL: http://wwwietforg/rfc/rfc2409txtl 1998;.

[15] Meadows C. Analysis of the Internet key Exchange Protocol using the NRL Protocol Analyzer. In Proceedings of IEEE Symposium on Security and Privacy, Oakland, CA,pp216 - 231 1999;.

[16] Zhou J. Further analysis of the Internet key exchange protocol. Computer Communications, Vol 23, No 17, pp1606–1612 2000;.

[17] Perlma R, Kaufman C. Analysis of the IPSec Key Exchange Standard. In Proceedings of Tenth IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, Cambridge, MA, USA, pp150–156 2001;.

[18] Aiello W, Bellovin SM, Blaze M, Canetti R, Ioannidis J, Keromytis AD, et al. Efficient, DoS Resistant, Secure Key Exchange for Internet Protocols. In Proceedings of the 9th ACM conference on Computer and communications security, Washington, USA, pp48–58 2002;.

[19] Haddad H, Berenjkoub M, Gazor S. A proposed protocol for Internet Key Exchange (IKE). In Proceedings of Electrical and Computer Engineering, Niagara Falls, Canada, pp2017–2020 2004;.

[20] Kaufman C. RFC 5996: Internet Key Exchange Protocol Version 2 (IKEv2). IETF, RFC 4306: Internet Key Exchange (IKEV2) Protocol', IETF, available at https://toolsietforg/html/rfc4306 2005;.

[21] Smith J, N GM, Boyd C. Modeling denial of service attacks on JFK with Meadows's cost-based framework. In Proceedings 4th Australasian Information Security Workshop, Hobart, Australia, pp125-134 2006;.

[22] Team T, et al. AVISPA v1. 1 User manual. Information society technologies programme (June 2006), http://avispa-project org 2006;.

[23] Kaufman C, Homan P, Nir Y, Eronen P. RFC 5996: Internet Key Exchange Protocol Version 2 (IKEv2). IETF, URL: http://wwwrfc-editororg/info/rfc5996 2010;.

[24] Zhu X, Haigang Z, Jun L. Analysis and Improvement of IKEv2 against Denial of Service Attack. In Proceedings of International Conference on Information, Networking and Automation (ICINA), Kunming, pp350–355 2010;.

[25] Nagalakshmi V, Rameshbabu I, Avadhani PS. Modified protocols for internet key exchange (IKE) using public encryption key and signature keys. In Proceedings of the eighth international conference on Information Technology: New Generations, Las Vegas, NV, pp376–381 2011;.

[26] Cremers C. Key Exchange in IPsec revisited: Formal Analysis of IKEv1 and IKEv2. Proceedings of 16th European Symposium on Research in Computer Security, Leuven, Belgium, pp315–334 2011;.

[27] Ray S, Nandan R, Biswas GP. ECC Based IKE Protocol Design for Internet Applications. In Proceedings 2nd International Conference on Computer, Communication, Control and Information Technology of Technology (Elsevier), pp522–529 2012;.

[28] Ahmim M, Babes M, Ghoualmi N. Formal analysis of efficiency and safety in IPSec based on internet key exchange protocol. IJCNDS, Vol14, No2, pp 202-218 2015;.

[29] Lavanya M, Natarajan V. Lightweight key agreement protocol for IoT based on IKEv. Computers and Electrical Engineering, Vol 64, pp 580-594 2017;.

[30] Glouche Y, Genet T, Heen O, Courtay O. A Security Protocol Animator Tool for AVISPA. In ARTIST2 Workshop on Security Specification and Verification of Embedded Systems, Pisa 2006;.

[31] Farash MS, Attari MA, Jami M. A new efficient authenticated multiple-key exchange protocol from bilinear pairings. Computers and Electrical Engineering, Vol 39, No 2, pp530–5 2013;.