International Journal of Information & Network Security (IJINS)
Vol.3, No.6, December 2014, pp. 353~358
ISSN: 2089-3299

Identity Based Authenticated Key Agreement for MANET

ShaheenaKhatoon*, Balwant Singh Thakur, Birendra Kumar Sharma***
*School of Studies in Mathematics, Pt.RavishankarShukla University Raipur (C.G.)
**School of Studies in Mathematics, Pt.RavishankarShukla University Raipur (C.G.)
*School of Studies in Mathematics, Pt.RavishankarShukla University Raipur (C.G.)

Article Info

Article history:

Received
Revised
Accepted

Keyword:

Key agreement
Pairings
MANET
Trusted authority
Identity based cryptography

ABSTRACT

A mobile ad-hoc network (MANET) is a convenient infrastructureless communication network. We can construct MANET on demand without support from central servers. MANETs can be applied in critical communication situations such as battlefield, emergency and rescue missions. In the mean time, MANET's have the following inherent characteristics: open medium, absence of fixed central structure, dynamically changing topology, constrained capabil- ity, etc, so MANETs are highly vulnerable to various security threats.To solve various security problem many key agreement protocols are presented. We de- scribe an authenticated key agreement (Ak) protocol by modifying Smart's AK protocol. We will also discuss the security of the protocol.

Corresponding Author:

ShaheenaKhatoon,
School of Studies in Mathematics, Pt.RavishankarShukla University Raipur (C.G.).
E-mail:shaheenataj.28@gmail.com

1. INTRODUCTION

Key agreement is process whereby two(or more) entities can establish a shared secret key. A key agreement protocol is said to provide implicit key authentication (of B to A) if A is assured that no other entity besides B can possibly ascertain the value of the secret key. A key agreement protocol that provides mutual implicit key au- thentication is called an authenticated key agreement protocol (or AK protocol).An authenticated key establishment protocol is called identity-based if users use their identity based asymmetric key pair, instead of a traditional public/private key pair, in the protocol for authentication and determination of the established key.

The first key agreement protocol based on asymmetric cryptography was the Diffie- Hellman protocol [2]. It is a fundamental technique providing unauthenti- cated key agreement using exponentiation. Its security is based on the intractability of the Diffie- Hellman problem and the discrete logarithm problem. Many key agree- ment protocols are based on the ideas of Diffie-Hellman, and such protocols can be described in any group in which the discrete logarithm is hard and exponentiation is efficient.

In 1984, Shamir [11] proposed the idea of using an identity based asymmetric key pair where an arbitrary string (typically an identity string) can be used as a user's public key. A trusted authority (TA) is required to derive private keys from arbitrary public keys. The TA also publishes public information required for all encryption, decryption, signature and verification algorithms in the system. This is referred to as identity-based cryptography (IBC). Shamir gave a practical ID-based signature scheme but left as an open question the problem of finding an efficient ID-based encryption scheme.

A few identity-based key agreement protocols have been developed based on Diffie- Hellman and using Shamir's key set up idea. For instances, Okamoto [10] presented an identity-based scheme and Tanaka and Okamoto slightly modify this in [14]. Girault and Pailles [5] developed an identity-based system, which can be used for non-interactive key agreement schemes.

In 2001 the first feasible solutions for identity-based encryption were published. One of them is Boneh and Franklin's identity-based encryption scheme [1], which is based on pairing on elliptic curves. Shortly after that, a few feasible identity-based key agreement protocols (as well as signature schemes) based on pairing techniques were developed. Smart, by combining the ideas from [1], [8] and [6], proposed an identity-based authenticated key agreement protocol (ID-AK) and an identity-based authenticated key agreement protocol with key confirmation (ID-AKC) in [13].

2. OUR CONTRIBUTION

The contributions of this paper are as follows:
1. Introducing an ID-AK protocol more efficient than Smarts.
2. Modifying Smart's and our proposed AK protocol to include the TA forward secrecy property and to avoid TAs being able to access user's communications.
3. Discussions on the security properties of these protocols.

3. PRELIMINARIES

3.1. Pairing Technique Concepts

Let G_1 and G_2 denote two groups of prime order q, where G_1, with an additive nota- tion, denotes the group of points on an elliptic curve; and G_2, with a multiplicative notation, denotes a subgroup of the multiplicative group of a finite field. A pairing is a computable bilinear map between these two groups. Two pairings have been studied for cryptographic use. They are Weil pairing [7, 12] and a modified version [1], and Tate pairing [3, 4]. For the purposes of this paper, we let e denote a general bilinear map, i.e., e: $G_1 \times G_1 \longrightarrow G_2$, which can be either a modified Weil pairing or a Tate pairing.

A Diffie-Hellman (DH) tuple in G_1 is (P, xP, yP, zP) $\in G_1$ for some x, y, z $\in Z_q^*$satisfying z = xymodq.Computational Diffie-Hellman (CDH) problem: Given any three elements from the four elements in a DH tuple compute the remaining element.CDH assumption: There exists no algorithm running in expected polynomial time, which can solve the CDH problem with non-negligible probability.Decision Diffie-Hellman (DDH) problem:GivenP,xP,yP,zP$\in G_1$ decide if it is a valid DH tuple.This can be solved in polynomial time by verifying e(xP, yP) = e(P, zP).Bilinear Diffie-Hellman (BDH) problem : Let P be a generator of G_1.The BDH problem in $\langle G_1, G_2, e\rangle$ is that given (P, xP, yP, zP) for some x, y, z $\in Z_q^*$compute $W = e(P,P)^{xyz} \in G_2$.BDH assumption: There exists no algorithm running in expected polynomial time, which can solve the BDH problem in $\langle G_1,G2,e\rangle$ with non-negligible probability.

Security of our authenticated key agreement protocols described in this paper is based on the CDH and BDH assumptions.

3.2 .Security Attributes

It is desirable for protocols to possess the following security attributes:

1. Known-key security: Each run of the protocol should result in a unique secret session key. The compromise of one session key should not compromise other session keys.
2. Forward secrecy: If long-term private keys of one or more of the entities are compromised, the secrecy of previously established session keys should not be affected. We say that a system has partial forward secrecy if the compromise of one (or more but not all) of the entities long-term keys can be corrupted without compromising previously established session keys, and we say that a system has perfect forward secrecy if the long-term keys of all the entities involved may be corrupted without compromising any session key previously established by these entities. There is a further (perhaps stronger) notion of forward secrecy in identitybased systems, which we call"TA forward secrecy", which certainly implies perfect forward secrecy. This is the idea that the TAs long-term private key may be corrupted (and hence all users long-term private keys) without compromising the security of session keys previously established by any users.
3. Key-compromise impersonation: The compromise of an entity as long- term private key will allow an adversary to impersonate A, but it should not enable the adversary to impersonate other entities to A.
4. Unknown key-share resilience: An entity A should not be able to be coerced into sharing a key with any entity C when in fact A thinks that she is sharing the key with another entity B.
5. Key control: Neither entity should be able to force the session key to be a preselected value.

4. SMART'S ID-BASED AK PROTOCOL

To describe the protocol, we use the notation, M_i : A $\longrightarrow$ B: m, to state that in the ith message flow, entity A sends a message m to entity B. This notation will be used throughout the paper. Smart's ID-AK protocol involves three entities: two users Alice and Bob who wish to establish a shared secret session key, and a TA from whom they each require their own private key.

The key generation center chooses a secret key,s ∈ {1, ..., i − 1}.The key generation centre produces a random P ∈ G and computes P_{KGS} = sP.Then the key generation centre publishes (P,P_{KGS}).When a user with identity ID wishes to obtain a public/private key pair's the public key is given byQ_{ID}=H(ID)، H is a hash function, H : {0, 1}*−→ G. The key generation centre computes the associated private key via S_{ID} = sQ_{ID}.

Authenticated Key Exchange: Suppose two users A and B wish to agree a key. We denote the private keys of these users by S_A =sQ_A andS_B =sQ_B, which have been obtained from the key generation centre. Each user generates an ephemeral private key, say a and b the data are then the values of the corresponding ephemeral public keys T_A = aP and T_B = bP as the following diagram shows

Table 1: Smart's Protocol

User A	User B
A T_A = aP K_{AB} = e(S_A; T_B)e(aQ_B; P_{KGS})	b T_B = bP K_{BA} = e(S_B; T_A)e(bQ_A; P_{KGS})

User A then computes, K_{AB} = e(S_A,T_B)e(aQ_B,P_{KGS}).User B then computes, K_{BA} = e(S_B,T_A)e(bQ_A,P_{KGS}).The secret key is then, K = V (K_{AB}) = V (K_{BA}), Where V is a key derivation function. This function will typically be a random oracle, or secure hash function. It is important to make use of these key derivation functions since an attacker might otherwise be able to gain partial information about the session key even though the underlying problem is hard.

This protocol has the following security properties: mutual implicit key authentication, known key security, partial forward secrecy, imperfect key control, key- compromise impersonation, and unknown key-share resilience.

We are now concerned about the following issues:

1. Efficiency. In Smart's protocol, each participant has to generate a random number, perform two elliptic curve point multiplications, and compute two pairings. In the next section, we will introduce a more efficient protocol, which offers the same security properties as Smart's Protocol.
2. Key escrow. As mentioned above, this protocol allows the TA to escrow the session key shared between Alice and Bob. This property may not acceptable for some applications. Although one main property of ID-based systems is that the TA generates private keys for users, some users may still want to conduct their own communications without the TA's eavesdropping. This security feature holds in the identity-based key agreement protocols using Shamir's key set up. In Section 5, we will describe a solution for Smart's protocol that avoids the TA being able to deduce the established key.

In the following sections, we give modifications of Smart's protocol each focusing on the above issues.

5. A MORE EFFICIENT AK PROTOCOL

We describe our first modification of Smart's Protocol. Alice and Bob each randomly choose an ephemeral private key, say a and b, then computes the corresponding ephemeral public keys W_A = aQ_A and W_B = bQ_B and exchange the public key as follows:

Table 2: Protocol 1

User A	User B
a W_A = aP K_{AB} = e(aS_A,W_B)	b W_B = bP K_{BA} = e(W_A, bS_B)

User A then computes K_{AB} = e(aS_A,W_B),User B then computesK_{BA} = e(W_A, bS_B),We first show that the secret shared keys agree,K_{AB} = e(aS_A,W_B)= e(asQ_A, bQ_B)= e(Q_A,Q_B)^{abs}= e(aQ_A,bsQ_B)= e(W_A, bS_B)= K_{BA}.The secret key is thenK = V (K_{AB}) = V (K_{BA}) = V (K).

Efficiency: The above protocol has same construction as Smart's Protocol. However, it is more efficient as Smart's Protocol requires each party to perform two elliptic curve point multiplications and two evaluations of the pairing. This protocol requires each party to perform two elliptic curve point multiplications and only one evaluation of the pairing.

5.1 Security Attributes

We now discuss some of the security properties related to these attacks.

1. Known key security: Protocol 1 has this property because each run of the protocol produces a different session key, therefore knowledge of past session keys does not allow deduction of future session keys.
2. Partial forward secrecy: We consider the following three separate parts of this property:

a. Compromise of long-term secret keys, either $S_A$orS_B, at some point in the future leads to the compromise of communications in the past, because K = e(aS_A,W_B) = e(W_A,bS_B). So the protocol does not offer perfect forward secrecy.
b. Compromising of the TA, s master key s leads to the compromise of com- munications in the past, because K = e $(aQ_A,bQ_B)^s$. This means the protocol does not offer TA forward secrecy.
c. Compromising of one or both of the ephemeral private keys, a and b, reveals none of the long-term secret keys,S_A,S_B and s, nor the shared secret session key V.
In the following section, we will modify Smart's Protocol and protocol 1 in order to provide TA forward secrecy, i.e., compromising the TAs master key (which also means compromising the two long-term secret keys of the users) does not lead to the compromise of communications in the past.

3. Imperfect key control: Protocol 1 does not have the full key control attribute since Bob can select his ephemeral key after having received Alice's ephemeral key. Bob can force l bits of the shared secret key to have a nominated value by evaluating K for roughly 2^l different choices of b. As it is noted in [9], the responder in a protocol almost always has an unfair advantage in controlling the value of the established session key. This can be avoided by the use of commitments, although this intrinsically requires an extra round.
4. Unknown key-share resilience. It is not easy to give a formal proof of whether the protocol possesses the unknown key-share resilience attribute or not. It seems to be difficult for an adversary to replace Alice's or Bob's public key with their own one because every user's public key is a hash functions output with their identity string as input. To give a formal proof of this security feature is an interesting open problem.
5. Key-compromise impersonation: When an adversary knows Alice's long-term private key, S_A, the adversary is not able to impersonate other entities, say Bob, to Alice.

6. Modification of Smart's Protocol and Protocol 1 without Key Escrow

There are two properties missing from Smart's Protocol and Protocol 1 that we may require;these are TA forward secrecy (and therefore also perfect forward secrecy), and the fact that we may not want the TA to be able to escrow session keys established by two users in the protocol.

Note that in identity-based cryptography (IBC) systems we cannot escape the possibility of a TA impersonating any user in the system because the TA is always able to do so. In PKI we have the same problem in fact a CA (certification authority) can generate a key pair, and (falsely) certify that the public key belongs to a user A. The CA can then impersonate A to any other user B. In both IBC and PKI we therefore have to assume that the trusted authority (TA or CA) will not impersonate users.

However a property that we may require from our identity-based key agreement protocol is that, if two users are actually communicating with each other (that is, no user is being actively impersonated by the TA), then the TA cannot derive (or therefore escrow) the established session key. This is mainly a privacy issue since users may trust the TA with their long-term keys, but may wish to be able to escape from the escrow environment (assuming no active attacks by the TA) for communications they wish to keep confidential even from the TA.

TA forward secrecy is another security issue. If at any stage the TAs key is compromised, this should not compromise the previously established session keys.

This is known as TA perfect forward secrecy, and in Protocols 1 and 2, this does not hold. Recall that the concept of perfect forward secrecy captures the idea that the corruption of both users long-term private keys does not compromise their pre- viously established session keys. Note that if the protocol has the property of TA forward secrecy, then it has perfect forward secrecy since the TA knows all users long-term private keys. The converse is not necessarily true since the TA knows not only all users long-term private keys, but also s, the TAs long-term master secret.

Ideally, we would like a key agreement protocol in which the long-term keys are used for authentication, but the ephemeral keys are used in a way that can- not be known by the TA or by anyone else who knows only the long-term secret keys. The most well-known method of achieving this is for the users to calculate a Diffie-Hellman shared key from their ephemeral contributions. We now introduce protocols modified from smart's Protocol and Protocol1, which have the above desired properties.

6.1 PROTOCOL 2

There seem to be a simple way to escape the key escrow using the Smart's protocol and Diffie-Hellman contributions. The key derivation function V in Smart's Protocol could be changed to$V' : G_2 \times G_1 \rightarrow \{0, 1\}^k$,and the shared secret key becomes K = V'(K, abP). We will refer to this protocol as Protocol 2. In this case, if an adversary compromises both the users' long-term private keys, S_A and S_B, at some point in the future, the adversary is not able to compromise communications in the past, because the adversary can calculate K but not abP. It is obvious that this modification also prevents the TA from being able to access the session key.

6.2 PROTOCOL 3

Note that this exact modification cannot be used in Protocol 3 because in Protocol 3, Alice and Bob exchange aQ_A and bQ_B, which are not Diffie-Hellman contributions. If avoidance of key escrow is required, we suggest the following modification. Let Alice and Bob exchange aQ_A, aP and bQ_B, bP. Then they compute K as the same as in Protocol 1. They finally compute the shared secret key as $K = V'(K, abP)$. We will refer to this protocol as Protocol 3. Compared with Protocol 2, Protocol 3 is more efficient on computation; in particular on pairing computations since only a single pairing is required, but less efficient on the message bandwidth since two points (as opposed to only one) need to be distributed by each user.

7. EFFICIENCY

The following table shows the efficiency comparison of our newly propose protocol with that of Smart's protocol.Ours protocol is more efficient then the Smart's protocol in term of evaluation of pairing, and offers same security properties as Smart's protocol.

Table 3: Comparison of efficiency

Protocol	Elliptic Curve Multiplications	Evaluation Of Pairing
Smart's Protocol	2	2
Our Protocol	2	1
Smart's Protocol without escrow	2	2
Our Protocol without escrow	2	1

8. CONCLUSION

We have investigated some security issues related to identity based authenticated key agreement, and proposed a few new protocols modified from Smart's protocol to efficiently achieve certain security properties.

ACKNOWLEDGEMENTS

The author wishes to thank the anonymous referees for their very useful comments and suggestions.

REFERENCES

[1] D. Boneh, and M. Franklin, "Identity-based encryption from the Weil pairing ," In Advances in Cryptology CRYPTO 01, LNCS 2139, pp. 213-229, Springer- Verlag, 2001.

[2] W.Diffie and M.E. Hellman, "New directions in cryptography," IEEE Transac- tions on Information Theory, vol. 22, pp. 644-654, 1976.

[3] G. Frey, M. Mller, and H. Rck, "The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems, " IEEE Transactions on Information Theory, vol. 45(5), pp.1717-1719, 1999.

[4] S.Galbraith, "Supersingular curves in cryptography, "In Advances in Cryptology Asiacrypt 01, LNCS 2248, pp. 495-513, Springer-Verlag, 2001.

[5] M.Girault and J.C. Paills, "An identity-based scheme providing zero-knowledge authentication and authenticated key exchange," In Proceeedings of ESORICS 90, pp. 173-184, 1990.

[6] A. Joux, "A one round protocol for tripartite Diffie-Hellman, " In Proceedings of Algorithmic Number Theory Symposium, ANTS-I , LNCS 1838, pp. 385- 394, Springer-Verlag, 2000.

[7] A.J.Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," IEEE Transactions on Information Theory, vol.39, pp.1639-1646, 1993.

[8] A.Menezes , M. Qu and S. Vanstone, "Some new key agreement protocols pro-viding mutual implicit authentication, " In Proceedings of the Second Workshop on Selected Areas in Cryptography (SAC '95, Ottawa, May 18-19), pp. 22-32.

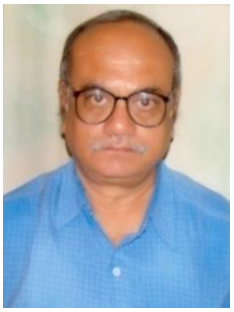
[9] C.Mitchell, M. Ward and P. Wilson, "Key control in key agreement protocols," Electronics Letters, vol.34, pp.980-981, 1998.

[10] E. Okamoto, "Proposal for identity-based key distribution system," Electronics Letters, vol.22, pp.1283-1284, 1986.

[11] A.Shamir, "Identity-based cryptosystems and signature schemes," In Advances in Cryptology - CRYPTO '84, LNCS 196, pp.47-53, Springer-Verlag, 1984.

[12] J.H.Silverman, "Advanced topics in the arithmetic of elliptic curves," GTM 151, ISBN 0-387-94325-0, Springer-Verlag, 1994.
[13] N.P. Smart, "An identity based authenticated key agreement protocol based on the Weil pairing," Electronics Letters, vol. 38, pp.630-632, 2002.
[14] K.Tanaka and E. Okamoto, "Key distribution system for mail systems using IDrelated information directory", Computers and Security, vol.10, pp.25-33, 1991.

BIBLIOGRAPHY OF AUTHORS (10 PT)

	Shaheena Khatoon received the B.Sc., M.Sc. and MPhil degree in Mathematics form Pt. Ravishankar Shukla University, Raipur. Chhattisgarh, India in 2005, 2007 and 2009. She joined School of Studies in Mathematics, Pt. Ravishankar Shukla University, Raipur, India for her research work.
	Balwant Singh Thakur Professor, School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur (C. G.) India. He has been working for long time in the field of Non Linear Operator Theory and currently in Cryptography. He and his research scholars work on many branches of public key cryptography. He is a life member of the Ramanujan Mathematical Society.
	Birendra Kumar Sharma Professor, School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur (C. G.) India. He has been working for long time in the field of Non Linear Operator Theory and currently in Cryptography. He and his research scholars work on many branches of public key cryptography. He is a life member of Indian Mathematical Society and the Ramanujan Mathematical Society.