# A Face Centered Cubic Key Agreement Mechanism for Mobile Ad Hoc Networks*

Ioannis G. Askoxylakis[1], Konstantinos Markantonakis[2], Theo Tryfonas[3], John May[3], and Apostolos Traganitis[1]

[1] Foundation for Reserach and Technology-Hellas – Institute of Computer Science, N. Plastira 100, 70013 Heraklion, Greece
{asko,tragani}@ics.forth.gr
[2] Royal Holloway University of London, Information Security Group, UK
K.Markantonakis@rhul.ac.uk
[3] University of Bristol, Faculty of Engineering, Queen's Building
University Walk, Clifton, Bristol, BS8 1TR
{t.tryfonas,j.may}@bristol.ac.uk

**Abstract.** Mobile ad hoc networking is an operating mode for rapid mobile node networking. Each node relies on adjacent nodes in order to achieve and maintain connectivity and functionality. Security is considered among the main issues for the successful deployment of mobile ad hoc networks (MANETs). In this paper we introduce a weak to strong authentication mechanism associated with a multiparty contributory key establishment method. The latter is designed for MANETs with dynamic changing topologies, due to continuous flow of incoming and departing nodes. We introduce a new cube algorithm based on the face-centered cubic (FCC) structure. The proposed architecture employs elliptic curve cryptography, which is considered more efficient for thin clients where processing power and energy consumption are significant constraints.

**Keywords:** MANET security, password authentication, elliptic curve cryptography, face-centered cubic (FCC) structure.

## 1 Introduction

Security is a primary concern for providing protected communications to mobile nodes that operate in hostile environments. Unlike the wireline networks or the mobile networks with hierarchical architecture like cellular networks, the unique nature and characteristics of Mobile Ad-hoc Networks (MANETs) pose a number of nontrivial challenges to security design, architecture and services. In MANETs nodes rely on each other in order to achieve and maintain connectivity and functionality.

* This work was supported in part by the European Commission in the 7th Framework Programme through project EU-MESH (Enhanced, Ubiquitous, and Dependable Broadband Access using MESH Networks), ICT-215320, http://www.eu-mesh.eu

F. Granelli et al. (Eds.): MOBILIGHT 2009, LNICST 13, pp. 103–113, 2009.

A MANET is a type of network, which is typically composed of equal mobile hosts that we call nodes. When the nodes are located within the same radio range, they can communicate directly with each other using wireless links. This direct communication is employed without hierarchical control. The absence of central control, such as base stations, introduces several problems, such as configuration advertising, discovery, maintenance, as well as ad hoc addressing, self-routing and security [1].

In this environment, trust cannot be provided among the nodes of the network without the existence of initial specific prior known information. This special kind of information is necessary in order to build trust between all participating nodes. An ad hoc network is established among the existing nodes, if from preexisting, commonly known information, we reach a state where a common Session Key is agreed among the nodes. Securing ad hoc networks is not trivial, mainly due to their dynamic topology and the vulnerability of the wireless links, which can be the medium for passive and active attacks.

In emergency situations or military operations the need for establishing a wireless network quickly and securely is crucial. The objective is to interconnect all computing and communication devices in a way that they will be able to share all necessary information securely, since nobody can guarantee that the "high tech" enemies will not try to disrupt or intercept the operation efforts.

The technical goal is to make sure that no other entity outside the *group* (*we define all the legitimate members of the established wireless network as group, e.g., soldiers of a military unit*) should be able to gain access within the new network. However, since neither a certification authority nor a secure communication channel exists, the enemy has the ability to eavesdrop and modify exchanged messages transmitted over the air. Additionally, since no central identification authority is present, group member impersonation is easy, jeopardizing the security of the whole system.

Considering all these issues, the main challenge that arises is the setting up of a wireless network where the legitimate members of a group will be able to establish a protected wireless network. Moreover, in the case where a new node arrives at place, desiring to become a member in an already established group, joining, without delaying or even intercepting the existing group, is also challenging. The case where a group member is captured by the enemy and therefore the group key is compromised is also part of the considered scenario.

## 2 Security Requirements

It is broadly known that security mechanisms cannot create trust [2]. The members of a team that wish to establish a MANET know and trust one another physically. Otherwise, they would never be able to achieve mutual trust regardless of the authentication mechanism used. Our goal is to exploit the existing physical mutual trust in order to secure the ad hoc network.

A password authentication mechanism seems to be a rational approach that can deliver a proper solution without adding new requirements like the use of dedicated hardware (i.e smart cards). In a password based authentication scheme the use of a sufficiently large and randomly generated data string that can be used as a password would be an obvious approach. This way all nodes could agree on a password and, by using a trivial authentication protocol, achieve mutual authentication.

In such a scenario, the underlying security depends on the size and the randomness of the chosen password. However, the larger the password gets the more difficult it is to memorize and use. Moreover, since the response time is vital during emergency operations, the use of large passwords can be proved inconvenient. Therefore the use of short, user-friendly passwords is an essential requirement.

The use of short passwords provides weak authentication since the password selection set is quite limited and thus the corresponding authentication procedure is vulnerable to dictionary attacks [3]. Therefore, we need an authentication protocol that will lead to a reasonable degree of security even if the authentication procedure has been initiated from a small, weak password.

Security threats can classified into two broad categories depending on their origin: external and internal attacks. External attacks originate outside the group while internal originate from already authenticated nodes belonging to the group. For instance, consider a group of soldiers operating in a hostile environment, trying to keep their presence and mission unknown to the enemy, and the case where a soldier, member of the mission group, is captured by the enemy who is now in a position to attack from inside. Another example, less extreme, is an ad hoc network formed in a classroom during a test exam between the laptops or PDAs of the students and the teacher's workstation. According to this scenario, not only we must secure the network from an external intruder but also from a student who temporarily exits the classroom in order to retrieve the solutions and then returns. In all cases, the misbehaving nodes must definitely be expelled from the established network.

At this stage it makes sense to outline the main security requirements of the proposed architecture:

*Weak-to-strong password-based authentication.* Use of an authentication scheme that will lead to a reasonable degree of security although the authentication procedure has been initiated from a small, weak password.

*Secure authentication.* Only the entities that hold the correct password will eventually become members of the MANET.

*Forward authentication.* Even if a malicious partner manages to compromise a network entity in a later phase, he will still be unable to participate in the already existing network.

*Contributory key establishment.* The MANET is established when a session key is generated and agreed among all network nodes. The session key should be generated throughout in a contributory manner, by all participating entities.

*Security architecture for thin clients.* A MANET is typically composed of mobile devices with limited processing power and energy consumption. The cryptographic algorithms used for authentication and key agreement should have minimal impact in terms of computational overhead.

The rest of the paper is organized as follows: In Section 3, we start with a review of the previous work concerning two-party and multiparty key agreements and we give a brief introduction on weak to strong authentication and the elliptic curve

theory. We describe the state of the art in multiparty key agreement protocols and particularly the d-cube and the body centered cubic algorithms and examine their properties. In Section 4, we propose a modification of the body-centered cubic algorithm, called face centered cubic algorithm designed for the dynamic changing topologies and we compare them. A discussion concerning the implementation issues and the problems that arise is presented in Section 5. Finally, in Section 6, we provide our concluding remarks along with suggestions for future work.

## 3 Related Work

### 3.1 Key Exchange and Elliptic Curve Cryptography

Common cryptographic protocols based on keys chosen by the users are weak to dictionary attacks. Bellowin and Merrit [4] proposed a protocol called *encrypted key exchange (EKE)* where a strong shared key is derived from a weak one. However, this protocol has a disadvantage. The creation of the common session key takes place with unilateral prospective, that is, only by the entity that first initiated the whole procedure. Thus the key agreement scheme is not contributory. In [5], Asokan and Ginzboorg proposed a contributory version of the above protocol for both two-party and multiparty cases.

Diffie–Hellman is the first public key distribution protocol that opened new directions in cryptography [5]. In this important protocol for key distribution, two entities *A, B* after having agreed on a prime number *p* and a generator *g* of the multiplicative group Z*p*, can generate a secret session key.

An essential property for the majority of cryptographic applications is the need for fast and precise arithmetic. Calculations over the set of real numbers are slow and inaccurate due to round-off error [6]. Finite arithmetic groups, such as

$$F_p, F_{2^m}.$$

which have a finite number of points, is used in practice. All practical public-key systems today exploit the properties of arithmetic using large finite groups. Additionally, elliptic curves can provide versions of public-key methods that, in some cases, are faster and use smaller keys, while providing an equivalent level of security. Consequently, the use of ECC can result in faster computations, lower power consumption, as well as memory and bandwidth savings. This is very useful for mobile devices, like the ones used in ad hoc networks, which face limitation in terms of CPU, power, and network connectivity.

An elliptic curve [7] consists of elements $(x, y)$ satisfying the equation:

$$y^2 = x^3 + \alpha x + \beta (\text{mod} p). \tag{1}$$

for two numbers $\alpha, \beta$. If $(x, y)$ satisfies the above equation then $P = (x, y)$ is a point on the elliptic curve.

The elliptic curve discrete logarithm problem (ECDLP) can be stated as follows:

Fix a prime $p$ and an elliptic curve $E$. Let $xP$ represent the point $P$ added to itself $x$ times. Suppose $Q$ is a multiple of $P$, so that $Q = xP$ for some $x$, then the ECDLP is to determine $x$ given $P$ and $Q$.

The general conclusion of leading cryptographers is that the ECDLP requires fully exponential time to solve. The security of ECC is dependent on the difficulty of solving the ECDLP.

Research community has given considerable attention to the ECDLP. Like the other types of cryptographic problems, no efficient algorithm is known to solve the ECDLP. The ECDLP seems to be particularly harder to solve. Moderate security can be achieved with the ECC using an elliptic curve defined over $Z_p$ where the prime $p$ is several times shorter than 230 decimal digits.

An elliptic curve cryptosystem implemented over a 160-bit field currently offers roughly the same resistance to attack, as would a 1024-bit RSA [8]. However, there have been weak classes of elliptic curves identified such as super singular elliptic curves [9] and some anomalous elliptic curves [10]. Implementations, such as ECDSA [11], merely check for weaknesses and eliminate any possibility of using these "weak" curves [12].

### 3.2 Elliptic Curve Diffie–Hellman

The original Diffie–Hellman (D-H) algorithm is based on the multiplicative group modulo $p$. However the elliptic curve Diffie–Hellman (ECDH) protocol is based on the additive elliptic curve group as desribed below. We assume that two entities $A, B$ have selected the underlying field, $GF(p)$ or $GF(2^k)$, the elliptic curve $E$ with parameters *a,b,* and the base point *P*. The order of the base point *P* is equal to *n*. Also, we ensure that the selected elliptic curve has a prime order to comply with the appropriate security standards [11].

At the end of the protocol, the communicating parties end up with the same value *K*, which represents a unique point on the curve. A part of this value can be used as a secret key to a secret-key encryption algorithm. We give a brief description of the protocol.

Entity *A* selects an integer,

$$d_A : d_A \in [2, n-2] \ . \tag{2}$$

Entity *B* selects an integer

$$d_B : d_B \in [2, n-2]. \tag{3}$$

*A* computes

$$Q_A = d_A \times P \ . \tag{4}$$

The pair $Q_A, d_A$ consists $A$ 's public and private key.

B computes

$$Q_B = d_B \times P . \tag{5}$$

The pair $Q_B, d_B$ consists $B$'s public and private key.

$A$ sends $Q_A$ to $B$,

$$A : Q_A \rightarrow B . \tag{6}$$

$B$ sends $Q_B$ to $A$,

$$B : Q_B \rightarrow A . \tag{7}$$

$A$ computes

$$K = d_A \times Q_B = d_A \times d_B \times P . \tag{8}$$

B computes

$$K = d_B \times Q_A = d_B \times d_A \times P . \tag{9}$$

Quantity $K$ is now the commonly shared key between $A$ and $B$. Moreover, it can also be used as a session key. Quantity $n$ is the order of the base point $P$.

### 3.3 D-Cube Protocols and Aggressive 3-D Cube Algorithm

For key establishment procedures in mobile ad hoc networks, where several entities are involved, multiparty authentication protocols should be applied. A lot of research has been done in this direction [13], [14]. Becker and Wille [15] presented a method very efficient in terms of number of authentication rounds. According to this method, also known as the d-cube protocol, all entities planning to participate in a network are initially arranged in a d-dimensional hypercube. Each potential network entity is represented as a vertex in the d-dimensional cube and it is uniquely assigned a d-bit address. The addresses are assigned in a way so that two vertices connected along the $i^th$ dimension differ only in the $i^th$ bit. There are $2^d$ vertices, each of which are connected to other d vertices.

In [17], a modified version of [16] called aggressive d-cube algorithm is presented, where faulty nodes are isolated from the ad hoc network during the early stages of the d-cube algorithm. According to the algorithm, the interaction of faulty-legitimate nodes and the chances a faulty node will enter the network by guessing the password are minimized. Moreover, their protocol protects legitimate nodes from unnecessary energy spending, which may be more important in case of thin clients.

To clearly demonstrate the differences between [17] and [16], we describe the algorithm of [17] through examples in 3-d case. In this case we assume that node G is the faulty partner. During the first round the DH key exchange procedure performed between (G:110) and (H:111) will fail, since node (G:110) is a faulty one. However, instead of remaining idle and wait for the next round (as in [16]), node (H:111) starts a DH key exchange with node (E:100). Meanwhile Node (E:100) has already performed a successful DH key exchange with (F:101), during the first half of the first round, so this key exchange will be the second successful one for this round. Node

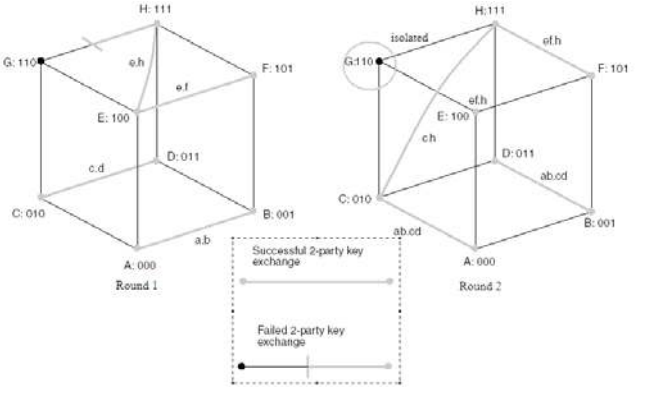


**Fig. 1.** Aggressive 3-d cube round 1 and 2

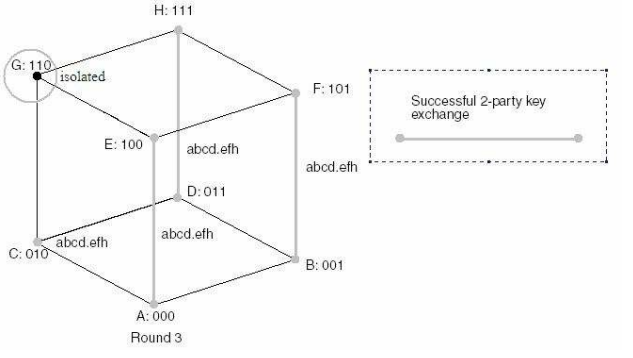


**Fig. 2.** Aggressive 3-d cube round 3

(E:100) having being notified by H that G is a faulty node will remain idle until the third round, instead of having attempted unnecessary DH exchanges with (G:110). In the next round (round 2) (H:111) performs a DH with node (F:101 ) and a DH with node (C:010 ). Given that (C:010 ) has performed two successful DH with (D:011) and (H:111) respectively, he will remain idle in the next round. However (C:010) has already performed a successful DH with (A:000), during round one.

In total node (C:010) has performed three successful DH, with three different nodes, which means that (C:010) has completed all the appropriate procedures. Thus it will remain idle for the next round, which is the last round in our case. Summarizing the description of this procedure, the upper bound of the total successful DH procedures for a node participating in an aggressive d-cube algorithm is equal to *d*. In this example $d = 3$. During the third and final round there will be three more successfully accomplished DH key exchanges. One between (H:111) and (D:011), one between (F:101 ) and (B:001 ), and one between (A:000) and (F:101).

Through this example it is clear that using the aggressive 3-d cube algorithm, the faulty partner is being isolated. He only participates in one DH key exchange, the one performed in round 1 with node (H:111), and since then he is excluded from all the

subsequent DH key exchanges. Consequently, the faulty node loses the ability to have another change, during the generation process of the common session key.

## 4 The Proposed Architecture

The dynamic topology of mobile ad-hoc networks introduces challenging security issues. The continuous flow of incoming and departing nodes is a key issue for designing a key agreement mechanism. Furthermore, when a node publicly claims that it is leaving the network it does not mean that it looses its ability to "hear" the messages exchanged among the remaining nodes, unless action is taken.

We propose a cryptographic key agreement algorithm that initiates from an aggressive 2-d or 3-d algorithm. The proposed method is a modification of [18], however it provides a completely different solution.

In the case of more than 8 nodes, instead of moving to a higher degree of space (4-d or more) we exploit the face centered architecture by arranging the next 6 new nodes on the centers of the 6 faces of the 3-d cube. For simplicity, in the rest of the paper, each bond in the 3-d space corresponds to a two-party, password based, elliptic curve, Diffie-Hellman key exchange as described in chapter 3.

### 4.1 The Face Centered Cubic (FCC) Algorithm

The proposed algorithm is depicted in figure 3. The first 8 nodes (or less) are arranged in a 3-d cube as shown in the left side of figure 3. They perform an aggressive 3-d cube algorithm and obtain a common session key. The first 6 nodes that will arrive in a later phase will be arranged in the centers of the six faces of the cube as shown in the central picture of figure 3.

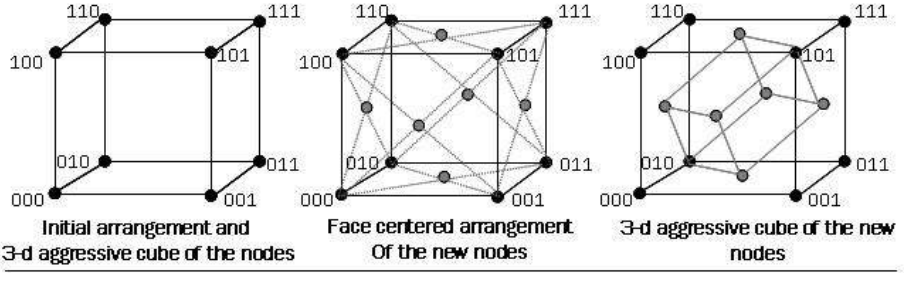


**Fig. 3.** The Face Centered Cubic algorithm

The 6 new nodes together with nodes (010) and (101) that contributed to the initial cube, create a new cube and perform a new (second) aggressive 3-d cube algorithm. This way the inner cube creates a second common session key. After the set up of the second session key nodes (010) and (101) hold both session keys corresponding to both cubes. This privilege makes nodes (010) and (101) leading nodes for the established network since any communication between black and grey nodes should pass through them. If we wish to avoid this hierarchy in our network, during the set up of

the common session key within the inner cube, nodes (010) and (101) propagate the common session key of the initial (black) cube to the new nodes. This way the first session key can be used by all nodes to communicate securely with each other, while the second can be used for the secure communication of the internal (grey) cube.

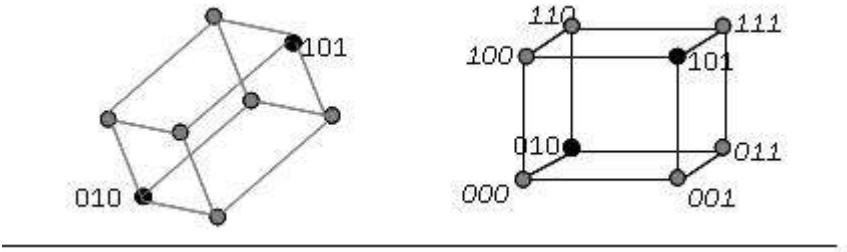


**Fig. 4.** Addressing of the new nodes

The addressing of the new nodes is shown in figure 4. We observe that the 2 old (black) nodes keep the same address with the one they had during the set up of the first session key. For the communication between nodes belonging to different cubes there is a separate metric (cube number) declaring the cube that the node is belonging to. In this example, black nodes are identified as cube 1 nodes, grey nodes are identified as cube 2 nodes and nodes (010) and (101) have both identifiers since they belong to both cubes.

### 4.2 Key Refreshment due to Departing Nodes

In the hierarchical model, where every cube has its independent session key, key refreshment due to departing nodes is easy. As soon as a node is leaving the network, the rest nodes of the common cube, perform a new aggressive 3-d cube algorithm and create a new session key. In case the leaving node is belonging to two consecutive cubes, a new aggressive 3-d cube algorithm is performed automatically to both cubes.

In the case where the previous session keys belonging to previous cubes are forwardly distributed to the next cubes, the key renewal should be performed to all previous cubes. This appears not to be a desired feature, since if there is a departure in the last cube all previous stages/cubes will be affected. However this can be also avoided if the set up is a combination of the two solutions. Periodically the key forwarding method is interrupted by the hierarchical solution. This way, we create isolated groups of concatenated cubes and any necessary key refreshment is bounded within these groups.

## 5 Conclusion

Our research was motivated from the requirement of certain groups to establish fast, reliable, efficient and secure MANET's without relying on pre-existing infrastructures. The actual operational environment and the very nature of the established networks impose further key issues (e.g. the ability to add or subtract nodes depending on operational and security considerations) that need to be taken into account.

We have reviewed existing proposals around two-party or multiparty authentication and introduced a new key establishment method. Our proposal overcomes some of the main issues (such as rapid deployment, accuracy, and dynamic and robust behaviour) of existing solutions and operational environments. The proposed solution introduces the use of elliptic curve cryptography in such a scenario. ECC computations require less storage, less power, less memory, and less bandwidth than other systems. This allows implementation of cryptography in constrained platforms such as wireless devices, handheld computers, smart cards, and thin-clients. For a given security level, elliptic curve cryptography raises computational speed and this is important in ad hoc networks, where the majority of the clients have limited resources.

We have also described known protocols for password authenticated multiparty DH key exchange and have chosen the aggressive cube algorithm due to its resilience against dictionary attacks. The proposed protocol meets all security requirements according the initial specification and it is stronger in terms of security. Finally, we have proposed a security architecture for dynamic MANETs, where the composition of the network changes in time with the arrivals and departures of nodes. The secure dynamic recomposition of the network could be proved very useful in battlefields where a soldier, under threat of capture, signs off the network on time.

The proposed FCC algorithm can be applicable in several other scenarios such as emergency situations, where rescue workers arrive at a disaster field, or for groups of people meeting in a room, i.e., in a classroom together with the teacher, etc. The password-based feature of our work could be used in cases where a group of people meets one another in person for the first time, and would like to go back home and set up a secure network among them.

The proposed algorithm leaves several open issues for future work. Formal analysis is necessary. The incorporation of several new password-based key agreement protocols, which do not require the use of asymmetric encryption, is a challenging consideration. The case where the number of network entities fluctuates unevenly, changing the network topology rapidly, is also very interesting.

## References

1. Verikoukis, C., Alonso, L., Giamalis, T.: Cross-Layer Optimization for Wireless Systems: A European Research Key Challenge. IEEE Communications Magazine 43(7), 1–3 (2005)
2. Bonnefoi, P.-F., Sauveron, D., Park, J.H.: MANETS: an exclusive choice between use and security? Special Issue on Interactive Multimedia & Intelligent Services in Mobile and Ubiquitous Computing (MUC) of COMPUTING AND INFORMATICS 27(5) (2008)
3. Narayanan, A., Shmatikov, V.: Fast dictionary attacks on passwords using time-space trade-off Conference on Computer and Communications Security. In: Proceedings of the 12th ACM conference on Computer and communications security, Alexandria VA USA (2005)
4. Bellovin, S.M., Merrit, M.: Encrypted key exchange: Password based protocols secure against dictionary attacks. In: Proceedings of the IEEE Symposium on Research in Security and Privacy, Oakland, USA (May 1992)
5. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Transactions on Information Theory 22, 644–654 (1976)

6. Cucker, F., Smale, S.: Complexity estimates depending on condition and round off error. Journal of the Association for Computing Machinery 46(1), 113–184 (2000)
7. Koblitz, N.: Elliptic curve cryptosystems. Mathematics of Computation 4(8), 203–209 (1987)
8. Rivest, R., Shamir, A., Adleman, L.M.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM 21(2), 120–126 (1978)
9. Menezes, A., Okamoto, T., Vanstone, S.: Reducing elliptic curve logarithms to logarithms in a finite field. IEEE Transactions on Information Theory 39, 1639–1646 (1993)
10. Menezes, A., Teske, E., Weng, A.: Weak Fields for ECC. In: Okamoto, T. (ed.) CT-RSA 2004. LNCS, vol. 2964, pp. 366–386. Springer, Heidelberg (2004)
11. Johnson, D., Vanstone, S.: The elliptic curve digital signature algorithm (ECDSA). International Journal on Information Security 1, 36–63 (2001)
12. Kalele, A.A., Sule, V.R.: Weak keys of pairing based Diffie-Hellman schemes on elliptic curves. Cryptology ePrint Archive 2005/30 (2005)
13. Zheng, D., Chen, K., You, J.: Multiparty authentication services and key agreement protocols with semi-trusted third party. Journal of Computer Science and Technology archive 17(6), 749–756 (2002)
14. Ateniese, G., Steiner, M., Tsudik, G.: New Multiparty Authentication Services and Key Agreement Protocols. IEEE Journal of Selected Areas in Communications 18(4) (April 2000)
15. Becker, C., Wille, U.: Communication complexity of group key distribution. In: 5th ACM Conference on Computer and Communications Security, San Francisco, California (November 1998)
16. Asokan, N., Ginzboorg, P.: Key agreement in ad hoc networks. Computer Communications 23, 1627–1637 (2000)
17. Askoxylakis, I.G., Kastanis, D.D., Traganitis, A.P.: Elliptic curve and password based dynamic key agreement in wireless ad-hoc networks, Communications. In: Networks and Information Security CNIS 2006, Cambridge, USA (October 2006)
18. Askoxylakis, I.G., Sauveron, D., Markantonakis, K., Tryfonas, T., Traganitis, A.: A Body-Centered Cubic Method for Key Agreement in Dynamic Mobile Ad Hoc Networks. In: Second International Conference on Emerging Security Information, Systems and Technologies, Cap Esterel, France, August 25-29, pp. 193–202 (2008)