Dynamic Topology Control In Manet's To Mitigate Spam Attacks Using Secure Communication Protocols.

[1,] N.NIMITHA , [2,]V.NIRMALA

[1,]M.E (P.G Student)-ECE, M.Kumarasamy college of Engineering
Karur, Tamilnadu, India

[2,]Lecturer-ECE, M.Kumarasamy college of Engineering
Karur, Tamilnadu, India

Abstract

Security is the main concern and bottleneck for widely deployed wireless applications due to the fact that wireless channels are vulnerable to attacks and that wireless bandwidth is a constrained resource. In this sense, it is desirable to adaptively achieve security according to the available resource. In particular, mobile ad hoc networks (MANETs) based on cooperative communication (CC) present significant challenges to security issues, as well as issues of network performance and management. This paper presents a secure decentralized clustering algorithm for mobile ad-hoc networks (SADTCA). The algorithm operates without a centralized controller, operates asynchronously, and does not require that the location of the modes be known a priori. Based on the cluster-based topology, secure hierarchical communication protocols and dynamic quarantine strategies are introduced to defend against spam attacks, since this type of attacks can exhaust the energy of mobile nodes and will shorten the lifetime of a mobile network drastically. By adjusting the threshold of infected percentage of the cluster coverage, our scheme can dynamically coordinate the proportion of the quarantine region and adaptively achieve the cluster control and the neighborhood control of attacks. Elliptic Curve Digital Signature Algorithm (ECDSA) which is one of the variants of Elliptic Curve Cryptography (ECC) proposed as an alternative to established public key systems such as Digital Signature Algorithm (DSA) has recently gained a lot of attention in industry and academia. The key generated by the implementation is highly secured and it consumes lesser bandwidth because of small key size used by the elliptic curves. In existing system, Joint topology control and authentication design in MANET's with cooperative communication scheme the spam attacks are not avoided, hence the throughput is minized. Simulation results show that the proposed approach is feasible and cost effective for mobile ad-hoc networks.

Keywords: *Secure communication protocols; adaptive topology control, MANETs, Cooperative communication, JATC topology control, quarantine region.*

I. INTRODUCTION

Mobile networks are typically characterized by limited power supplies, low bandwidth, small memory sizes and limited energy [1, 2].Thus the resource-starved nature of Mobile networks poses great challenges for security, since wireless networks are vulnerable to security attacks due to the broadcast nature of the transmission medium [2–15].In most Mobile network applications, the lifetime of Mobile nodes is an important concern, which can shorten rapidly under spam attacks. Moreover, maintaining network connectivity is crucial to provide reliable communication in wireless ad-hoc networks. In order not to rely on a central controller, clustering is carried out by adaptive distributed control techniques. To this end, the Secure Adaptive Distributed Topology Control Algorithm (SADTCA) aims at topology control and performs secure self-organization in four phases: (I) Anti-node Detection, (II) Cluster Formation, (III) Key Distribution; and (IV) Key Renewal, to protect against spam attacks.In Phase I, in order to strengthen the network against spam attacks, the secure control is embedded into the SADTCA. A challenge is made for all Mobiles in the field such that normal nodes and anti-nodes can be differentiated. In Phase II, based on the operation in Phase I, the normal Mobiles may apply the adaptive distributed topology control algorithm (ADTCA) from [16] to partition the Mobiles into clusters. In Phase III, a simple and efficient key distribution scheme is used in the network. Two symmetric shared keys, a cluster key and a gateway key, are encrypted by the pre-distributed key and are distributed locally.

A cluster key is a key shared by a clusterhead and all its cluster members, which is mainly used for securing locally broadcast messages. Moreover, in order to form a secure inter-cluster communication channel, a symmetric shared key may be used to encrypt the sending messages between the gateways of adjacent clusters. Since using the same encryption key for extended periods may incur a cryptanalysis risk, in Phase IV, key renewing may be necessary for protecting the Mobile network and guarding against the adversary getting the keys.Built upon the cluster-based network topology, three quarantine methods, Method 1: quarantine for clusters, Method 2: quarantine for nodes, and Method 3: quarantine for infected areas, are proposed for dynamically determining the quarantine region. In order to explore the fundamental performance of the SADTCA scheme, an analytical discussion and experiments are presented to investigate the energy consumption, communication complexity, the increase of communication overheads for data dissemination, and the percentage of the quarantine region in the sensing field when facing the spam attack.The organization of this paper is as follows: In Section 2., we briefly introduce the related work and summary of security issues for wireless Mobile network environment. Section 3. describes the system model and algorithm for secure self-organization in a cluster-based network topology. Section 4. presents dynamic approaches for determining the quarantine region. In Section 5., we analyze the SADTCA and make comparisons with protocols in the flat-based topology. In Section 6., the simulation results are shown and discussed. Finally, Section 7. draws conclusions and shows future research directions.

II. LITERATURE REVIEW

There are many vulnerabilities and threats to wireless Mobile networks. The broadcast nature of the transmission medium incurs various types of security attacks. Different schemes to detect and defend against the attacks are proposed in [2–15]. A number of anti-nodes deployed inside the sensing field can originate several attacks to interfere with message transmission and even paralyze the whole Mobile network. Most network layer attacks against Mobile networks fall into one of the following categories:

Acknowledgement Spoofing.
Selective Forwarding.
Sybil attacks.
Wormholes attacks.
Sinkhole attacks.
Hello flood attacks.

The spam attack, which is a kind of flooding Denial of Service (DoS) attack, can be carried out by the anti-node inside the Mobile network. Such attack can retard the message transmission and exhaust the energy of a Mobile node by generating spam messages frequently.

2.1 SECURE ADAPTIVE DISTRIBUTED TOPOLOGY CONTROL ALGORITHM

In this section we present a secure adaptive distributed topology control algorithm (SADTCA) for wireless Mobile networks. The proposed algorithm organizes the Mobiles in four phases: Anti-node Detection, Cluster Formation, Key Distribution, and Key Renewal. The main keys used in the network are (a) Pre-distributed Key, (b) Cluster Key, and (c) Gateway Key. Each Mobile is pre-distributed with three initial symmetric keys, an identification message, and a key pool. Pre-distributed key is established with key management schemes [6, 7], and is used for anti-node detection and cluster formation in Phases I and II. The Cluster Key and Gateway Key are used for key distribution in Phase III. The key pool is used for key renewing in Phase IV. Note that since our research aims at network topology control, the pre-distributed key establishment is beyond the scope of this paper.

2.2 Phase I:Anti-Node detection:

In order to strengthen the network against spam attacks, the secure control is embedded into the SADTCA. An authenticated broadcasting mechanism, such as the μTESLA in SPINS [20], may be applied in this phase. In the authenticated broadcasting mechanism, a challenge is made for all Mobiles in the field such that normal nodes and anti-nodes can be differentiated. The challenge is that when a Mobile broadcasts a H ello message to identify its neighbors, it encrypts the plaintext and then broadcasts; when receiving the H ello message, the Mobile decrypts it. If the Mobile decrypts the received message successfully, the sender is considered normal. Otherwise, the sender is said to be an anti-node. Therefore, we keep on the network topology without anti-nodes in order to make the network safe.If an anti-node is presented in the first deployment of a Mobile network, its neighboring normal nodes will notice the existence of the anti-node, since the anti-node will fail in authentication. Thus, referring to the cluster-based topology formed in Phase II, the spam attacks can be handled by adaptively forming the quarantine region as detailed in Section 4.Notice that an external attack can be prevented by the operation of Phase I. In this work, we do not have a lightweight countermeasure to defend against authenticated malicious nodes. If the authenticated node is compromised and performs malicious activities, a mechanism for evicting the compromised nodes is required [7].

2.3 Phase II: Cluster Formation

When Mobiles are first deployed, the adaptive distributed topology control algorithm (ADTCA) from [16] may be used to partition the Mobiles into clusters. The following subsections overview the mechanisms of the ADTCA scheme for cluster formation.

2.4 Cluster head Selection:

Each Mobile sets a random waiting timer, broadcasts its presence via a ‘Hello’ signal, and listens for its neighbor’s ‘Hello.’ The Mobiles that hear many neighbors are good candidates for initiating new clusters; those with few neighbors should choose to wait. By adjusting randomized waiting timers, the Mobiles can coordinate themselves into sensible clusters, which can then be used as a basis for further communication and data processing.Mobiles update their neighbor information (i.e., a counter specifying how many neighbors it has detected) and decrease the random waiting time based on each ‘new’ Hello message received. This encourages those Mobiles with many neighbors to become clusterheads. Therefore, if the timer expires, then the Mobile declares itself to be a clusterhead, a focal point of a new cluster. However, events may intervene that cause a Mobile to shorten or cancel its timer. For example, whenever the Mobile detects a new neighbor, it shortens the timer. On the other hand, if a neighbor declares itself to be a clusterhead, the Mobile cancels its own timer and joins the neighbor’s new cluster.After applying the ADTCA, there are three different kinds of Mobiles: (1) the clusterheads (2) Mobiles with an assigned cluster ID (3) Mobiles without an assigned cluster ID, which will join any nearby cluster after τ seconds and become 2-hop Mobiles, where τ is a constant chosen to be larger than all of the waiting times. In this phase, each Mobile initiates 2 rounds of local flooding to its 1-hop neighboring Mobiles, one for broadcasting Mobile ID and the other for broadcasting cluster ID, to select clusterheads and form 2-hop clusters. Hence, the time complexity is O(2) rounds. Thus, the topology of the ad-hoc network is now represented by a hierarchical collection of clusters.

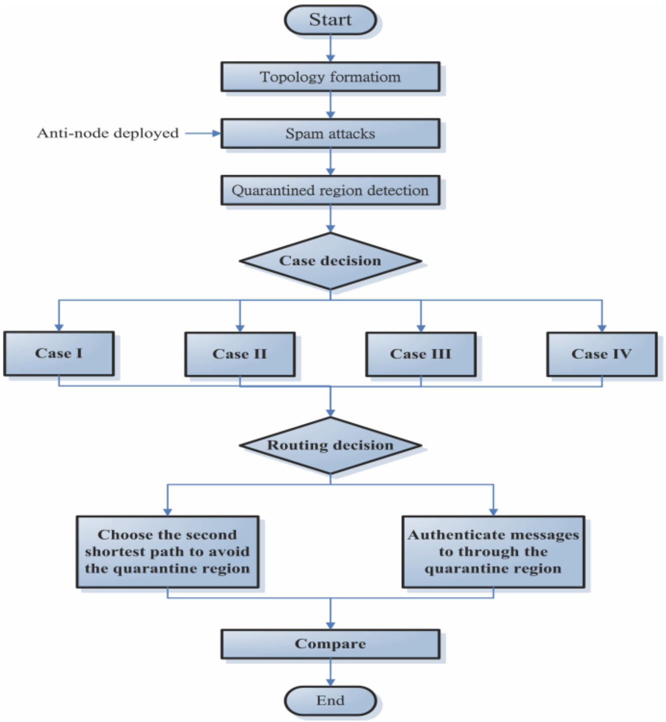
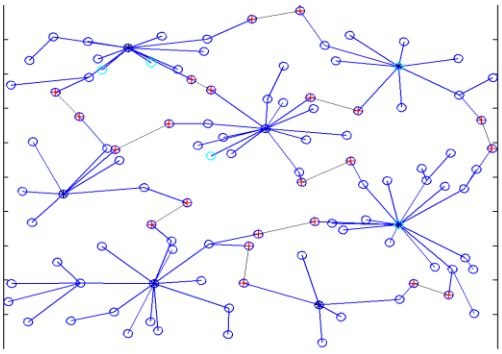

Start
Topology formatiom
Anti-node deployed
Spam attacks
Quarantined region detection
Case decision
Case I
Case II
Case III
Case IV
Routing decision
Choose the second shortest path to avoid the quarantine region
Authenticate messages to through the quarantine region
Compare
End

Figure 1. Simulation flowchart of a SADTCA



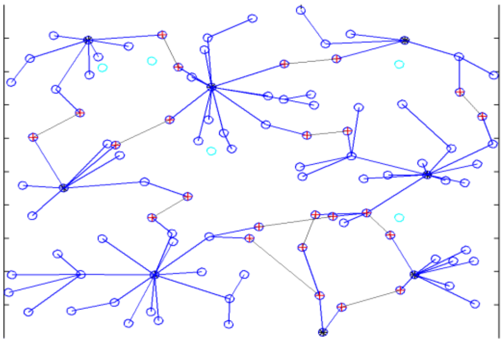


Figure 2a.The Mobile network without secure topology control Figure 2b.The Mobile network with secure topology control.

2.5 Phase III: Key Distribution:

According to the cluster construction in Phase II, a simple and efficient key distribution scheme is applied in the network. In this phase, two symmetric shared keys, a cluster key and a gateway key, are encrypted by the pre-distributed key and are distributed locally. A cluster key is a key shared by a clusterhead and all its cluster members, which is mainly used for securing locally broadcast messages, e.g., routing control information, or securing Mobile messages. Moreover, in order to form a secure communication channel between the gateways of adjacent clusters, a symmetric shared key may be used to encrypt the sending message. The process of key distribution is shown in Figure 3. In this phase, another challenge may be made to guard against anti-nodes that have not been found out in Phase I. The challenge is that if any Mobile cannot decrypt ciphertext encrypted by a cluster key or a gateway key, the node will be removed from the member or neighbor list. Therefore, the security of intra-cluster communication and inter-cluster communication are established upon a cluster key and a shared gateway key, respectively.

2.6 Phase IV: Key Renewal:

Using the same encryption key for extended periods may incur a cryptanalysis risk. To protect the Mobile network and prevent the the adversary from getting the keys, key renewing may be necessary. In the case of the revocation, in order to accomplish the renewal of the keys, the originator node generates a renewal index, and forwards the index to the gateways. The procedures of key renewal are detailed as follows.Initially all clusterheads (CHs) choose an originator to start the "key renewals", and then it will send the index to all clusterheads in the network. There are many possible approaches for determining the originator. For instance, the clusterhead with the highest energy level or the clusterhead with the lowest cluster ID. After selecting the originator, it initializes the "Key renewal" process and sends the index to its neighboring clusters by gateways. Then the clusterhead refreshes the two keys from the key pool and broadcasts the two new keys to their cluster members locally. The operation repeats the way through to all clusters in the network. The key renewing process is depicted in Figure 3. A period of time (Tr) is set in order to avoid that the originator does not start the "key renewal" process. If the other clusters do not receive the index after Tr , they will choose a new originator from themselves. The method helps to rescue when the previous originator is broken off.

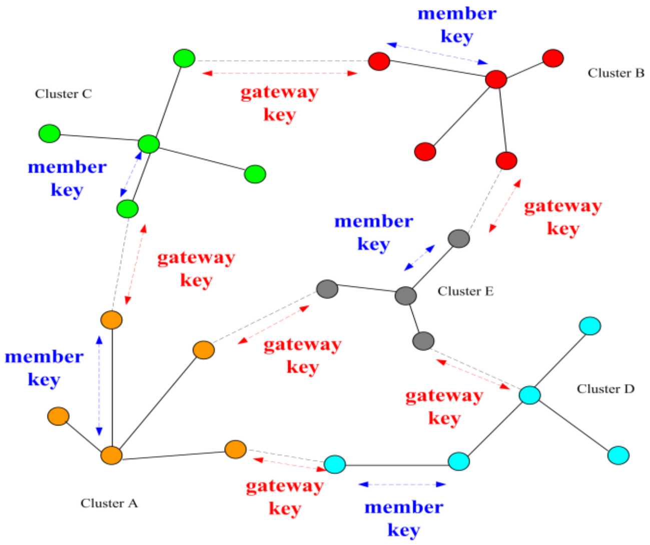


Figure 3 .Key distribution

III. ELLIPTIC CURVE DIGITAL SIGNATURE ALGORITHM

Elliptic Curve Digital Signature Algorithm is implemented over elliptic curve P-192 as mandated by ANSI X9.62 in C language. The Project contains necessary modules for domain parameters generation, key generation, signature generation, and signature verification over the elliptic curve.ECDSA has three phases, key generation, signature generation, and signature verification.

3.1 ECDSA Key Generation:

An entity A's key pair is associated with a particular set of EC domain parameters D= (q, FR, a, b, G, n, h). E is an elliptic curve defined over Fq , and P is a point of prime order n in E(Fq), q is a prime. Each entity A does the following:

[1] Select a random integer d in the interval [1, n- 1].

[2] Compute Q = dP.

[3] A's public key is Q, A's private key is d.

3.2 ECDSA Signature Generation:.

To sign a message m, an entity A with domain parameters D= (q, FR, a, b, G, n, h) does the following:

[1] Select a random or pseudorandom integer k in the interval [1, n-1].

[2] 2. Compute kP =x1, y1 and r= x1 mod n (where x1 is regarded as an integer between 0 and q-1). If r= 0 then go back to step 1.

[3] Compute k^{-1}mod n.

[4] Compute s= k^{-1} {h (m)+ dr} mod n, where h is the Secure

[5] Hash Algorithm (SHA-1). If s = 0, then go back to step 1.

[6] The signature for the message m is the pair of integers (r, s).

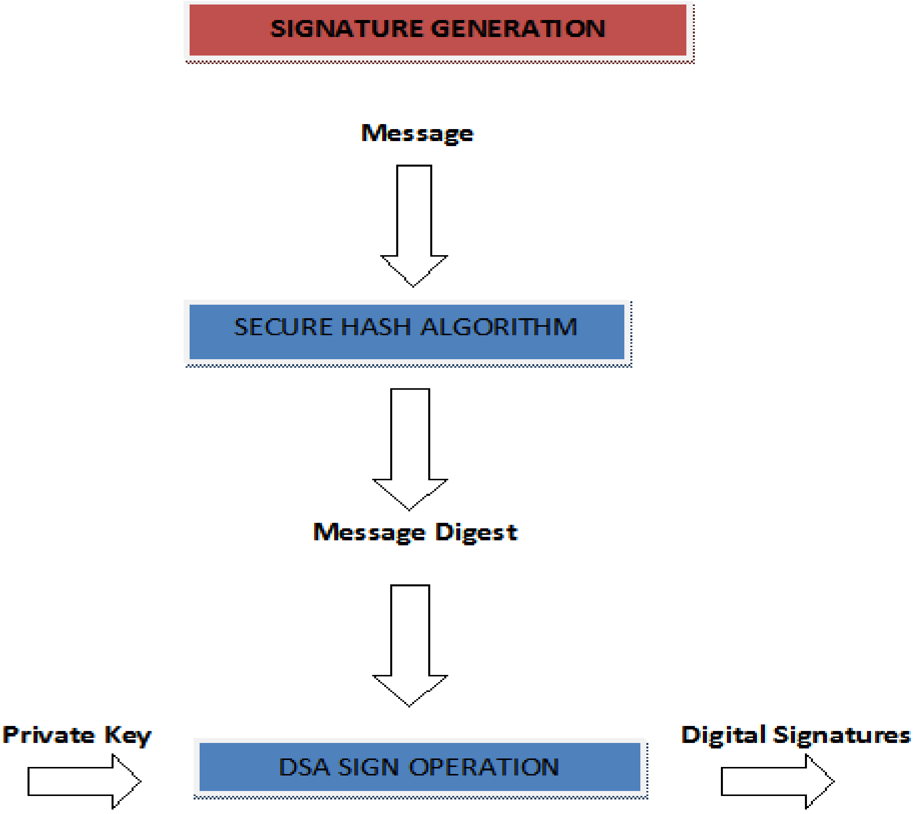


Table 1 Key comparison of Symmetric, RSA/DSA/DH, ECC

3.3 ECDSA Signature Verification:

To verify A's signature (r, s) on m, B obtains an authenticated copy of A's domain parameters D = (q, FR, a, b, G, n, h) and public key Q and do the following

[1] Verify that r and s are integers in the interval [1, n-1].

[2] Compute w = s^{-1}mod n and h (m)

[3] Compute u1 = h(m)w mod n and u2 = rw mod n.

[4] Compute u1P + u2Q =(x0, y0) and v= x0 mod n.

[5] Accept the signature if and only if v = r

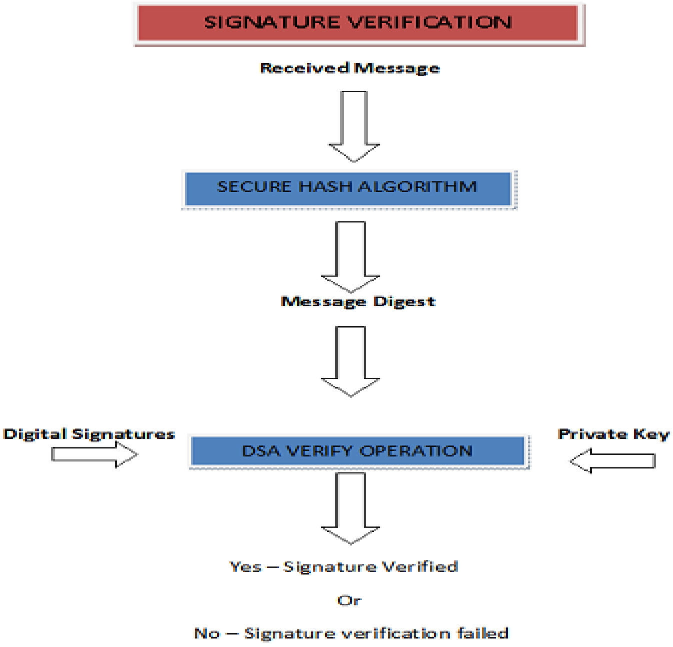
[6] Comparison of ECDSA with DSA

[1] Both algorithms are based on the ElGamal signature scheme and use the same signing equation: s = k^{-1}\{h (m) + dr\} mod n.

[2] In both algorithms, the values that are relatively difficult to generate are the system parameters(p, q and g for the DSA; E, P and n for the ECDSA).

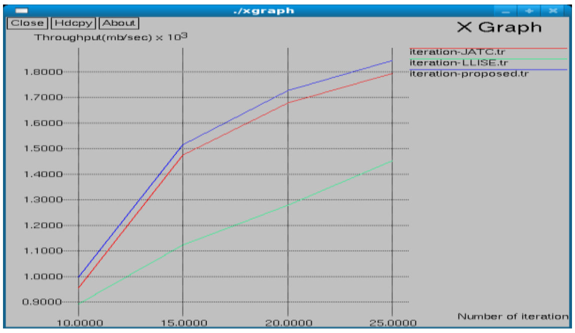
[3] In their current version, both DSA and ECDSA use the SHA-1 as the sole cryptographic hash function.

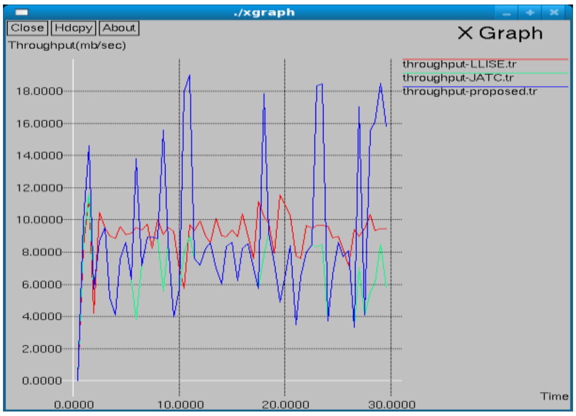
[4] The private key d and the per-signature value k in ECDSA are defined to be statistically unique and unpredictable rather than merely random as in DSA [11].



Symmetric	RSA/DSA/DH	ECC	Time to break in MIPS
80	1024	160	10^{12}
112	2048	224	10^{24}
128	3072	256	10^{28}
192	7680	384	10^{47}
256	15360	512	10^{66}

Table 1 Key comparison of Symmetric, RSA/DSA/DH, ECC





Aggregate throughput and digital signature in the Aggregate throughput using elliptic curve algorithm with respect to authentication protocol versus different numbers of nodes time

IV. CONCLUSION

We describe a secure protocol for topology management in wireless Mobile networks. By adaptively forming quarantine regions, the proposed secure protocol is demonstrated to reach a network security agreement and can effectively protect the network from energy-exhaustion attacks. Therefore, in a hierarchical network topology, the SADTCA scheme can adapt cluster control and neighborhood control in order to achieve dynamic topology management of the spam attacks. Compare to the JATC scheme, the spam attacks are avoided and the throughput is increased. The main reason for the attractiveness of ECDSA is the fact that there is no sub exponential algorithm known to solve the elliptic curve discrete logarithm problem on a properly chosen elliptic curve. Hence, it takes full exponential time to solve while the best algorithm known for solving the underlying integer factorization for RSA and discrete logarithm problem in DSA both take sub exponential time. The key generated by the implementation is highly secured and it consumes lesser bandwidth because of small key size used by the elliptic curves. Significantly smaller parameters can be used in ECDSA than in other competitive systems such as RSA and DSA but with equivalent levels of security. Some benefits of having smaller key size include faster computation time and reduction in processing power, storage space and bandwidth. This makes ECDSA ideal for constrained environments such as pagers, PDAs, cellular phones and smart cards. These advantages are especially important in other environments where processing power, storage space, bandwidth, or power consumption are lacking.

Although the initial secure goals of the research have been achieved in this paper, further experimental and theoretical extensions are possible. In our future work, we plan to involve more mechanisms to make the protocol faultless and practical, such as developing a new algorithm to identity anti-network sensors and proposing efficient security mechanisms to make protocol suitable for adaptive topology management.

REFERENCES

[1] Al-Karaki, J.N.; Kamal, A.E. Routing techniques in wireless sensor networks: A survey. IEEE Wirel. Commun. 2004, 11, 6–28.

[2] Djenouri, D.; Khelladi, L. A survey of security issues in mobile ad hoc and sensor networks. IEEE Commun. Surv. Tutor. 2005, 7, 2–28.

[3] Karlof, C.; Wagner, D. Secure routing in wireless sensor networks: attacks and countermeasures. Ad Hoc Netw. 2003, 1, 293–315.

[4] Karlof, C.; Sastry, N.; Wagner, D. TinySec: A link layer security architecture for wireless sensor networks. In Proceedings of the 2nd international conference on Embedded networked sensor systems, Baltimore, MD, USA, November 3-5, 2004; ACM New York, NY, USA, 2004; pp. 162-175.

[5] Yi, S.; Naldurg, P.; Kravets, R. A security-aware routing protocol for wireless ad hoc networks. In Proceedings Of ACM Symposium On Mobile Ad Hoc Networking & Computing (MOBIHOC), Lausanne, Switzerland, June 9-11, 2002; pp. 286-292.

[6] Zhu, S.; Setia, S.; Jajodia, S. LEAP: efficient security mechanisms for large-scale distributed sensor

[7] networks. In Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS' 03), Washington, DC, USA, October 27-30, 2003; pp. 62-72.

[8] Dimitriou, T.; Krontiris, I. A localized, distributed protocol for secure information exchange in sensor networks. In Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, Denver, Colorado, April 3-8, 2005; p. 240a.

[9] Li, H.; Singhal, M. A secure routing protocol for wireless ad hoc networks. In Proceedings of the 39th Hawaii International Conference on System Sciences, Kauai, HI, USA, January 4-7, 2006; Volume 9, pp. 225a.

[10] Wood, A.; Stankovic, J. Denial of service in sensor networks. Computer 2002, 35, 54–62.

[11] Wood, A.; Stankovic, J.; Son, S. JAM: A jammed-area mapping service for sensor networks. In Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS'03), Cancun , Mexico, December 3-5, 2003; pp. 286-297.

[12] Newsome, J.; Shi, E.; Song, D.; Perrig, A. The sybil attack in sensor networks: Analysis & defenses. In Proceedings of the third international symposium on Information processing in sensor networks, Berkeley, CA, USA, April 26-27, 2004; pp. 259-268.

[13] Hu, Y.C.; Perrig, A.; Johnson, D.B. Wormhole detection in wireless ad hoc networks. In Rice University Department of Computer Science Technical Report TR01-384, Rice University, Houston, TX, USA, 2002.

[14] Anderson, R.; Kuhn, M. Tamper resistance - a cautionary note. In Proceedings of the Second Usenix Workshop on Electronic Commerce, Oakland, California, November 18-21, 1996; pp. 1-11.

[15] Roosta, T.; Shieh, S.; Sastry, S. Taxonomy of security attacks in sensor networks and countermeasures. In Proceedings of The First IEEE International Conference on System Integration and Reliability Improvements, Hanoi, Vietnam, December, 2006; pp. 13-15.

[16] Shaikh, R.A.; Jameel, H.; d'Auriol, B.J.; Lee, H.; Lee, S.; Song, Y.-J. Group-based trust management scheme for clustered wireless sensor networks. IEEE Trans. Parall. Distrib. Sys. 2009, 20, 1698–1712.

[17] Chu, K.-T.; Wen, C.-Y.; Ouyang, Y.-C.; Sethares, W. A. Adaptive distributed topology control for wireless ad-hoc sensor networks. In Proceedings of 2007 International Conference on Sensor Technologies and Applications (SENSORCOMM 2007), Valencia, Spain, October 14-20, 2007; pp. 378-386.

[18] Sancak, S.; Cayirci, E.; Coskun, V.; Levi, A. Sensor wars: detecting and defending against spam attacks in wireless sensor networks. In Proceedings of IEEE International Conference on Communications, Paris, France, June 20-24, 2004; pp. 3668-3672.

[19] Coskun, V.; Cayirci, E.; Levi, A.; Sancak, S. Quarantine region scheme to mitigate spam attacks in wireless sensor networks. IEEE Trans. Mobil. Comput. 2006, 5, 1074–1086.

[20] Wen, C.-Y.; Sethares, W. A. Automatic decentralized clustering for wireless sensor networks.EURASIP J. Wirel. Commun. Netw. 2005, 5, 686–697.

[21] Perrig, A.; Szewczyk, R.; Tygar, J. D.; Wen, V.; Culler, D. E. SPINS: Security protocols for sensor networks. Wirel. Netw. 2002, 8, 521–534.

[22] Santi, P. Topology Control in Wireless Ad Hoc and Sensor Networks; John-Wiley & Sons: Chichester, UK, 2005.

[23] Quansheng Guan,Member, IEEE, F. Richard Yu, Senior Member, IEEE, Shengming Jiang,Senior Member, IEEE,and Victor C. M. Leung,Fellow, IEEEJoint Topology Control and Authentication Designin Mobile Ad Hoc Networks WithCooperative Communications, IEEE transactions on vehicular technology , vol. 61, no. 6, july2012