Multi-Layer Key Establishment for Large Scale Sensor Networks

Panayiotis Kotzanikolaou*
Department of Informatics,
University of Piraeus, GR-18534, Greece
E-mail: pkotzani@unipi.gr
*Corresponding author

Dimitris D. Vergados and Giannis Stergiou
Department of Information and Communication Systems Engineering,
University of the Aegean, Karlovassi, Samos, GR-832 00, Greece
E-mail: {vergados,gstergiou}@aegean.gr

Emmanuil Magkos
Department of Computer Science,
Ionian University, GR-49100, Corfu, Greece
E-mail: emagos@ionio.gr

Abstract: Research on key establishment for Distributed Sensor Networks (DSNs) focuses on lightweight protocols that are feasible for the sensor nodes, which by default have restricted capabilities. Although the most efficient protocols for key establishment are based on symmetric key encryption, these protocols are unable to provide adequate security against attacks, such as node impersonation and fake generation attacks. For this reason, several hybrid key establishment protocols have been developed, making limited use of public key cryptography, and more particularly of Elliptic Curve Cryptography. However, although these protocols seem to be efficient for sensor nodes, they reduce performance, especially in large-scale networks. In this paper, we propose a multi-layer key establishment protocol for DSNs, which combines hybrid and symmetric key establishment techniques. The performance analysis shows a reasonable decrease in performance, due to the optimized use of expensive public-key cryptographic operations.

Keywords: Distributed Sensor Networks, Security, Key Establishment

Reference to this paper should be made as follows: Kotzanikolaou, P., Vergados D., Stergiou G., and Magkos E. (2005) ‘Multi-Layer Key Establishment for Large Scale Sensor Networks’, *Int. J. Security and Networks*, Vol. 1, Nos. 1/2/3, pp.43–54.

Biographical notes: P. Kotzanikolaou was born in 1974. He received his BSc in Computer Science in 1998 from the University of Piraeus, Greece and his PhD in 2003 from the same university. His research focuses on information and communications security and applied cryptography for mobile agent systems and distributed systems, as well as for networks such as wireless ad hoc networks, intelligent networks and sensor networks.

D.D. Vergados was was born in Athens, Greece in 1973. He is a Lecturer in the University of the Aegean, Department of Information and Communication Systems Engineering. He received his B.Sc. in Physics from the University of Ioannina and his Ph.D. in Integrated Communication Networks from the National Technical University of Athens, Department of Electrical Engineering and Computer Science. His research interests are in the area of Communication Networks (Wireless Broadband Networks, Sensor - Ad-hoc Networks, Military Networks, WLANs, IP, MIP, SONET Networks), Neural Networks, GRID Technologies, and Computer Vision. He has participated in several projects funded by EU and National Agencies. He is also Guest Editor and Reviewer in several Journals and he has served in technical program committees of several international conferences.

G. Stergiou is a PhD Candidate in the University of the Aegean, Department of Information and Communication Systems Engineering. He received his B.Sc. in Mathematics from the University of Crete and his M.Sc. in Technologies and Management of Information and Communication Systems Engineering from the University of Aegean, Department of

Information and Communication Systems Engineering. His research interests are in the area of Information and Communication Systems Security, mainly in Wireless Networks (WLAN, Ad hoc Networks and Sensor Networks) and in the area of Grid Technologies. Giannis Stergiou is a student member of the IEEE.

E. Magkos was born in 1975. He obtained his degree in Computer Science (1997) and his PhD (2003) entitled "Secure electronic transactions over the Internet" by the Department of Informatics at the University of Piraeus, Greece. He is currently teaching security in the Computer Science Department, Ionian University, Corfu, Greece. His research interests include information security and cryptography, wireless networks, distributed systems

1 INTRODUCTION

Distributed Sensor Networks (DSN) are becoming increasingly popular as they can be used in a variety of civil applications such as temperature monitoring, motion, moisture and light sensing, as well as natural disaster control and health care. Of special interest is the usage of DSNs in secure-critical domains: thousands of such nodes could be deployed in unattended and/or adversarial environments to collect information such as tracking hostile troop movements, or detecting chemical and biological weapons.

DSNs are in general more vulnerable to security threats than other wired or wireless networks. The sensor nodes are not physically protected and adversaries with radio equipment may eavesdrop communications, modify packets, inject system with false data or prevent routing of messages. Moreover, sensors may be “captured” due to physical or remote attacks and an adversary might obtain all stored information, including keying material and private sensor readings. Depending on the environment where nodes are deployed, appropriate protection measures should be taken for data confidentiality, integrity and authentication between communicating entities, while taking into account the cost, storage, energy and communication efficiency requirements. To support such security services one needs key management techniques that are resilient to both external and internal attacks. The required trade-off makes it an important challenge to design secure communications for DSNs.

In several applications of DSNs, it may be required that the network is updated with other nodes in future time periods, in order to extend the network or replace erroneous nodes. Each set of incoming nodes that will join the network in a future time consists of a *node generation*. The nodes of a particular generation are pre-deployed with the appropriate keys, which will enable them to perform key bootstrapping with each other, as well as with nodes of a previous generation. The protocols which allow multiple key bootstrapping phases between nodes of different generations are known as *multi-phase* deployment protocols.

An interesting research area is establishing secure communication channels between pairs of sensor nodes in self-organising networks (Eschenauer and Gligor, 2004), *i.e.* networks that do not rely on any fixed infrastructure. In a typical self-organizing DSN all the sensor nodes will engage simultaneously in a bootstrapping phase and will exchange pairwise keys. Since the nodes in DSNs have very limited energy, computation and storage resources, key establishment techniques for self-organising DSNs must be very efficient. Obviously, protocols based on low-cost symmetric cryptography are more appropriate, compared to the expensive public-key cryptography. For this reason, several lightweight key establishment protocols for DSNs have been developed, such as (Deturtre et al., 2004) and (Zhu et al., 2003). Although such protocols offer adequate security for civil applications, they are not appropriate for highly sensitive applications of sensor networks, since they are vulnerable to several attacks, such as impersonation attacks during the key bootstrapping period (Kotzanikolaou et al., 2005a).

Although pure asymmetric (public-key) key establishment protocols are not efficient for sensor nodes, recent research has shown that it is feasible to employ limited public-key cryptography through hybrid protocols that use Elliptic Curve Cryptography (erticom Research, 2000) techniques – see for example (Malan et al., 2003; Gaubatz et al., 2004; Huang et al., 2003; Kotzanikolaou et al., 2005a). These protocols may be more resilient to impersonation attacks than the symmetric ones, since they allow each node to be uniquely identified in a cryptographic way. Furthermore, they allow multiphase key deployment where nodes joining the network in a future time period are allowed to establish keys with the existing nodes of the previous generation(s). Moreover, some of these protocols prevent nodes belonging to a certain generation to impersonate nodes belonging to another generation, (*e.g.* Kotzanikolaou et al., 2005a), an attack known as *fake generation attack*. Unfortunately, although these hybrid key establishment protocols are efficient for DSNs, they are still more expensive than the symmetric protocols.

In this paper, we propose a multi-layer key establishment scheme, which combines hybrid and symmetric key establishment techniques. The proposed scheme has three layers of key establishment. The nodes are clustered in geographic areas and the nodes belonging in the same neighborhood use a symmetric key establishment protocol in order to exchange pairwise keys. Furthermore, in each cluster there exists one node with extended capabilities. This extended node is able to communicate with other nodes of its category, located in the surrounding clusters. These nodes exchange keys by using the hybrid key establishment protocol proposed in (Kotzanikolaou et al., 2005a). Finally, these nodes can be used to assist the ordinary nodes belonging to distant areas to securely exchange pairwise keys. Moreover, the use of the hybrid protocol allows for secure multi-phase deployment of sensor nodes. The performance analysis of the proposed scheme shows a reasonable increase in performance, since the hybrid protocol is not extensively used.

This paper is organized as follows: The following section discusses the key establishment protocols proposed in the literature. Section 3 and 4 present the proposed multi-layer key establishment scheme for sensor networks and the Security of the multi-layer scheme respectively. The system Model is discussed in section 5 and the Performance analysis of the proposed system is taking place. Finally, the paper's conclusions are presented in the last section.

2 RELATED WORK

Symmetric key establishment protocols seem more suitable for DSNs, due to the constrained resources of the sensor nodes. In (Dutertre et al., 2004) and (Zhu et al., 2003), two symmetric-key approaches are proposed for secure multiphase deployment. In these schemes all nodes belonging to a certain node generation *i* are pre-deployed with a set of symmetric keys, which are used for key establishment. These protocols are very efficient for key establishment in DSNs and provide adequate security for several uses. However, in the protocols of (Dutertre et al., 2004) and (Zhu et al., 2003), it is assumed that the nodes cannot be attacked during the key bootstrapping phases. This may be a strong assumption for highly sensitive uses of DSNs, *e.g.* military or disaster recovery uses, since the sensors cannot be tamper-resistant due to their physical limitations.

Moreover, since only symmetric encryption techniques are employed, the nodes cannot prove their participation in a specific node generation. This could be useful in some circumstances, *e.g.* when a fresh node must be programmed to cooperate with sensors of a specific generation, or when inter-generation communication shall be given higher priority. In such cases, the above protocols could be subject to fake generation attacks (Kotzanikolaou et al., 2005a), where corrupted nodes may pretend to belong to another node generation than the actual one.

For such cases, hybrid key establishment protocols have been developed, which make use of limited public key cryptography, and more particularly Elliptic Curve Cryptography (ECC)– see for example (Certicom, 2000, Gaubatz et al., 2004; and Huang et al., 2003). Huang et al., (2003) propose a hybrid key establishment protocol for pairwise key establishment in DSNs, by combining ECC and symmetric encryption. To minimize the number of the expensive scalar multiplications, the authors Huang et al., (2003) propose the employment of some Full-Functional Devices (FFDs) that will take most of the cryptographic burden. The cost for each restricted sensor node is then reduced to one scalar multiplication with a random point and one scalar multiplication with a static point, per key establishment. This cost seems to be tolerable for security-critical DSNs.

In (Kotzanikolaou et al., 2005a) a hybrid key establishment protocol for sensor networks is proposed. The protocol combines standard Elliptic Curve Diffie-Hellmann (ECDH) key establishment with symmetric encryption techniques.

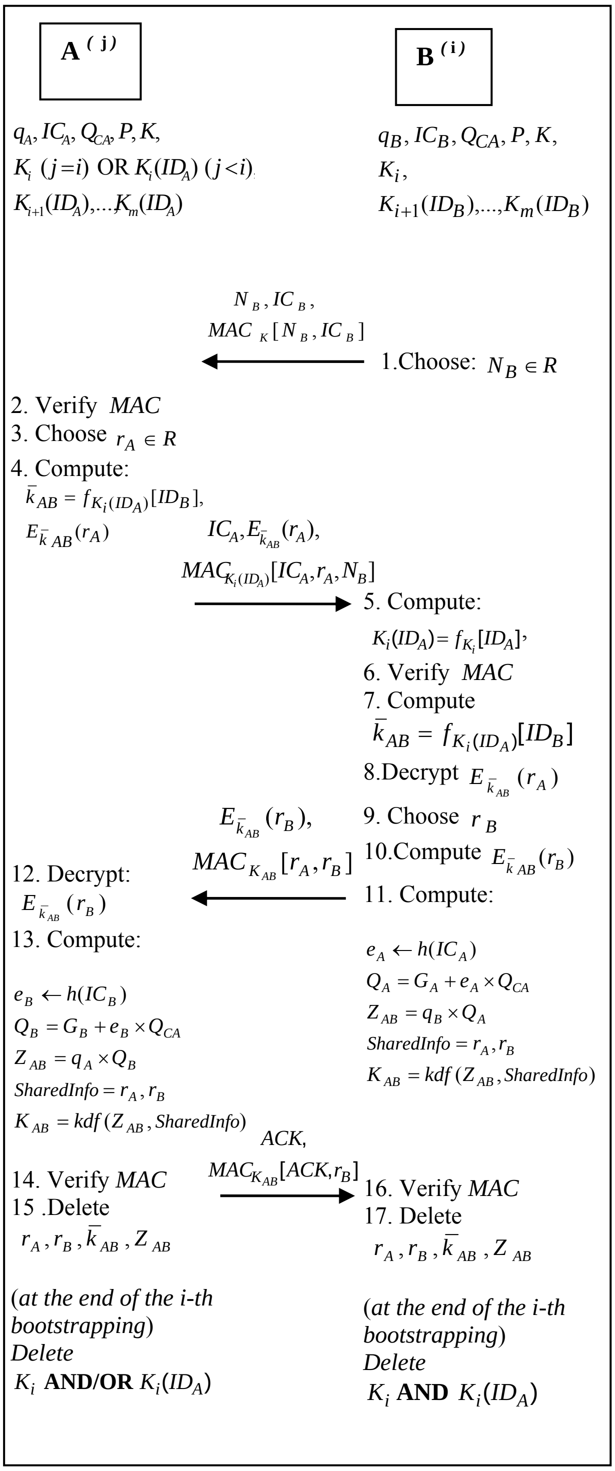
$A^{(j)}$
$B^{(i)}$
$q_A, IC_A, Q_{CA}, P, K,$
K_i $(j=i)$ OR $K_i(ID_A)$ $(j<i)$
$K_{i+1}(ID_A),...K_m(ID_A)$
$q_B, IC_B, Q_{CA}, P, K,$
$K_i,$
$K_{i+1}(ID_B),...,K_m(ID_B)$
$N_B, IC_B,$
$MAC_K[N_B, IC_B]$
1.Choose: $N_B \in R$
2. Verify MAC
3. Choose $r_A \in R$
4. Compute:
$\bar{k}_{AB} = f_{K_i(ID_A)}[ID_B],$
$E_{\bar{k}_{AB}}(r_A)$
$IC_A, E_{\bar{k}_{AB}}(r_A),$
$MAC_{K_i(ID_A)}[IC_A, r_A, N_B]$
5. Compute:
$K_i(ID_A) = f_{K_i}[ID_A],$
6. Verify MAC
7. Compute
$\bar{k}_{AB} = f_{K_i(ID_A)}[ID_B]$
8.Decrypt $E_{\bar{k}_{AB}}(r_A)$
9. Choose r_B
10.Compute $E_{\bar{k}_{AB}}(r_B)$
$E_{\bar{k}_{AB}}(r_B),$
$MAC_{K_{AB}}[r_A, r_B]$
11. Compute:
12. Decrypt:
$E_{\bar{k}_{AB}}(r_B)$
13. Compute:
$e_A \leftarrow h(IC_A)$
$Q_A = G_A + e_A \times Q_{CA}$
$Z_{AB} = q_B \times Q_A$
$SharedInfo = r_A, r_B$
$K_{AB} = kdf(Z_{AB}, SharedInfo)$
$e_B \leftarrow h(IC_B)$
$Q_B = G_B + e_B \times Q_{CA}$
$Z_{AB} = q_A \times Q_B$
$SharedInfo = r_A, r_B$
$K_{AB} = kdf(Z_{AB}, SharedInfo)$
$ACK,$
$MAC_{K_{AB}}[ACK, r_B]$
14. Verify MAC
15 .Delete
$r_A, r_B, \bar{k}_{AB}, Z_{AB}$
16. Verify MAC
17. Delete
$r_A, r_B, \bar{k}_{AB}, Z_{AB}$
(*at the end of the i-th bootstrapping*)
Delete
K_i **AND/OR** $K_i(ID_A)$
(*at the end of the i-th bootstrapping*)
Delete
K_i **AND** $K_i(ID_A)$

Figure 1 *The key establishment phase of the hybrid protocol proposed by Kotzanikolaou et al., (2005a)*

The authentication of the EC keys is based on Implicit Certificates, issued by an off-line Certification Authority. The computation cost of the EC cryptographic actions for each sensor is reduced to a scalar multiplication over a static point and a scalar multiplication over a random point. The cost reduction is due to the combination of symmetric encryption in the randomization process and the use of EC-Schnorr signatures (Shnorr, 1991) in the Implicit Certificate verification. The protocol is scalable, with sensors being pre-deployed with a constant number and size of keys, regardless of the size of the network.

The key establishment phase of this protocol between to nodes A and B is shown in Figure 1. Here, Q_{CA} denotes the public key of an off-line Certification Authority, while q_A, Q_A, and IC_A denote the Elliptic Curve secret-public key pair and the Identity Certificate of node A. Furthermore, $K_i(ID_A)$, $K_{i+1}(ID_A)$,…, $K_m(ID_A)$ denote the symmetric key of a node A which will be used during the key establishment phase of a particular node generation i, $i+1$, …,m. Note that these symmetric pre-deployed keys have the same structure as the pre-deployed keys in the symmetric scheme of Zhu et al., (2003).

The protocol of (Kotzanikolaou et al., 2005a) improves over the symmetric-key based schemes proposed in (Dutertre et al., 2004) and (Zhu et al., 2003), as it does not allow a compromised node to impersonate other nodes, belonging to the same or a different generation. Furthermore, it provides forward secrecy both in respect to a particular node and a generation of nodes. Moreover, it does not require the assumption of a protected bootstrapping period, although if such a protection exists the security of the protocol is further increased. Finally, it improves over the hybrid scheme of (Huang et al., 2003), since it supports multiphase deployment, and does not require the existence of full-functional devices.

3 THE PROPOSED MULTI-LAYER KEY ESTABLISHMENT SCHEME FOR SENSOR NETWORKS

Although the key establishment protocol of Kotzanikolaou et al., (2005a) is feasible for sensor nodes as analyzed in Kotzanikolaou et al. (2005b), it is still more expensive than the symmetric key establishment schemes. In order to maintain the advantages of the hybrid protocols such as forward secrecy, improved security against impersonation and fake generation attacks, multi-phase deployment and in order to maintain a more tolerable performance decrease, we propose a multi-layer key establishment scheme. This scheme uses both the hybrid and the symmetric protocols. The symmetric protocol is used for the majority of the nodes. The hybrid protocol is used only by few selected nodes, which have more power, communication and communication resources than the rest of the nodes. These nodes with the extended capabilities will be used for indirect authentication of the plain nodes to each other, as well as for the communication with nodes of future generations. Bellow, we describe the proposed scheme in detail.

Set up. We describe a multi-layer key establishment scheme for wireless sensor networks. The key establishment is performed in three phases, which are sequentially executed one after the other. The protocol assumes that the clusterheads are pre-deployed with the appropriate keying material, *i.e.* the symmetric and elliptic curve keys and the identity certificates, required by the hybrid key establishment protocol of Kotzanikolaou et al., (2005a) the plain nodes are pre-deployed only with symmetric keying material, as required by the protocol of Zhu et al, (2003). All the key pre-deployement has been performed off-line, before the initiation of the network.

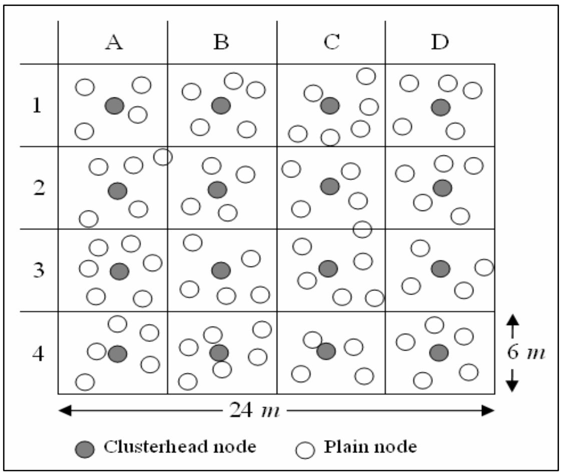

Figure 2 *A network consisting of 16 clusters of (6m× 6m) area*

We assume that the network is divided into logical clusters, for example square regions of a limited range. We also assume that the network consists of two types of nodes, the *clusterheads* and the *plain nodes*. Each cluster contains one node with extra computation, communication and power capabilities, which is defined as the clusterhead node. The clusterhead is placed approximately at the center of the cluster and it is capable to communicate with the clusterhead nodes of all its neighboring clusters. Each cluster also contains a number of plain nodes, which are the ordinary sensor nodes with limited capabilities. The communication range of the plain nodes is restricted approximately inside the cluster they belong.

Furthermore, we assume that after their deployment, all the nodes in the network have a fixed location, *i.e.* they are not mobile nodes. Finally, we assume that before the initiation of the key establishment scheme, all the nodes (the clusterheads and the plain sensor nodes) have established an appropriate routing protocol. Since the nodes are fixed and do not change location, a table-driven routing protocol may be more efficient.

An instance of the network described above comprised of 16 clusters is shown in Figure 2. Each cluster is defined as a $(6m,6m)$ square. It contains a clusterhead node, as well as a number of plain nodes. A plain node may be located in the border of a cluster and thus may be in direct range with more than one clusterheads. However, each plain node belongs to a unique cluster. For simplicity we assume that each node responds only to the first clusterhead hello message it receives.

After the deployment of the nodes and the establishment of the routing protocol, the key establishment scheme is initiated in three sequential phases. In the first phase, each clusterhead node establishes a key with all its neighboring clusterheads. In the second phase, all the nodes contained in a cluster establish a pairwise key with the other nodes lying within the same cluster. The first two phases are proactively performed by all the nodes. Finally, in the third phase, plain nodes of one cluster are able to establish a key with a clusterhead or a plain node of another cluster. The third phase is reactively performed only by plain nodes requesting to communicate securely with remote nodes. Note that the proposed key establishment scheme can be used for multi-phase deployment, where nodes are grouped to generations of nodes and each generation can join the network in a future time period and perform key establishment. We assume that each forthcoming node generation of nodes also contains clusterhead nodes and plain nodes, pre-deployed accordingly with the appropriate keys.

The following phases of key establishment will take place each time a forthcoming node generation joins the network, *i.e.* during each new key bootstrapping period. Each phase of the key establishment scheme creates a key establishment layer. The first phase generates a key establishment layer between the clusterheads. The second phase of the scheme generates a key establishment layer inside each cluster. Finally, the third phase uses the two former layers in order to generate a key establishment layer between distant plain nodes and clusterheads.

3.1 1st phase – Key establishment between clusterheads

In this phase the clusterheads perform key establishment with the clusterheads of the neighboring clusters, by using the hybrid protocol of Kotzanikolaou et al., (2005a). The hybrid protocol is executed for a given time only by the clusterheads, during which the rest of the nodes do not participate. This can be implemented by programming the plain sensors to initialize their communication after the time required by this phase. In the scenario presented in Figure 2, each clusterhead will perform bootstrapping with its one-hop surrounding clusters. For example, the clusterhead of the cluster B2 will perform the bootstrapping with the clusterheads of the squares A1, B1, C1, A2, C2, A3, B3 and C3.

3.2 2nd phase – Key establishment inside clusters

After the previous phase where each clusterhead has securely established a key with its neighboring clusterheads, the second phase of the scheme takes place. Each plain node in each cluster will perform a key establishment with its clusterhead, as well as with other plain nodes within its cluster, by using the symmetric protocol of Zhu et al., (2003). Recall that all the sensor nodes are already pre-deployed with the appropriate keying material. Each plain node *X* of the current node generation *i*, will perform key establishment with all neighboring nodes, by using their appropriate generation key K_i (according to the protocol of Zhu et al., (2003), key establishment is performed with an instance of the generation-wide key K_i that is linked to a specific node X with identifier ID_X, i.e. the key K_i (ID_X)). The cost of each key establishment in this phase is the cost of the symmetric protocol and it is restricted by the number of pairs of nodes in each cluster. After the end of this phase, each plain node will share a security association with the plain nodes inside the cluster it belongs, as well as with its clusterhead. Each plain node will share a pairwise key with one clusterhead node. In case where a plain node is in range with more than one clusterheads, it will be programmed to respond only to the response of the first clusterhead.

Note that in this phase all the nodes inside a cluster belonging to the *i*-th node generation will perform a key establishment with all other nodes of the same generation that lie within the same cluster.

3.3 3rd phase –Key establishment outside clusters

Although there exists a secure communication layer between neighbouring clusters from the first phase through the clusterheads, and a local secure communication layer from the second phase, there is still a need for secure communication between distant nodes. This communication is established in the third phase. In this phase, a plain node belonging in a given cluster may perform a key establishment with a node outside its cluster. The target node may either be a clusterhead node of another cluster or a plain node of another cluster. The key establishment will be performed with the symmetric protocol of Zhu et al., (2003) as in the previous phase. However, since the end nodes are out of range, the messages that must be exchanged between them in order to establish a key will be securely relayed through the path of intermediate clusterhead nodes. Figure 3 presents some examples of key establishment between remote nodes. The end nodes (the source and the destination) will be mutually authenticated along the path in a point-to-point manner until the destination node. The exchange of the key establishment messages will be secured by using the secure links established in the previous phases. The authentication and the encryption for message exchange within a given cluster is based on the pairwise keys derived from the 2nd phase, while the authentication and encryption for message exchange between intermediate clusterheads is based on the pairwise keys derived from the 1st phase. In Figure 3, three examples of key establishment are presented, for source and destination nodes of several distances. The destination node can be either a plain node or a clusterhead node.

Observe that in this phase, the key establishment messages are encrypted and decrypted in a point-to-point manner, throughout the chain of the source node, the intermediate clusterheads and the target node. This adds the cost of the intermediate encryption-decryption process in the communication chain from the source node to the destination node and vice versa, during the key establishment protocol. The above paradigm can include as many intermediate clusterheads are necessary to support the key establishment of any pair of nodes. Of course, as the number of intermediate nodes increases, the cost of point-to-point encryption also increases. Thus, there must be a balance in the number of intermediate nodes in order to be feasible for large-scale deployment. After the third phase is over, all nodes belonging to the generation *i* will delete their generation-wide keys K_i and all the instance keys.

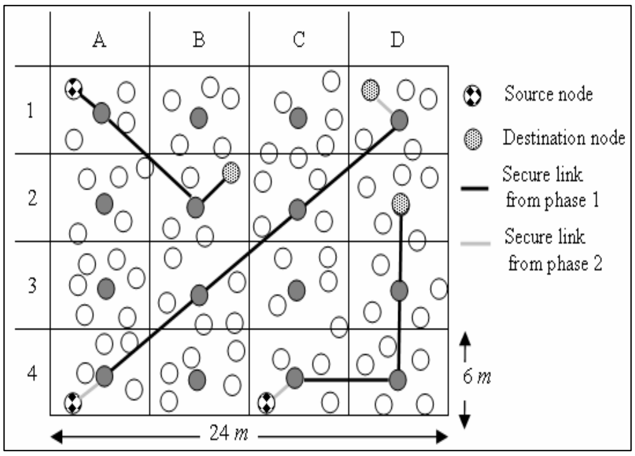

Figure 3 *Key establishment between nodes of different clusters (3rd phase)*

In the above paradigm, both nodes assumingly belong to the same node generation *i*. However, the key establishment scheme can also support the bootstrapping between nodes belonging to different node generations, by using the appropriate pre-deployed keys in a straightforward manner. This is possible since any node *X*, either a plain node or a clusterhead node, is pre-deployed with the symmetric generation keys $K_i(ID_X)$, $K_{i+1}(ID_X)$,…, $K_m(ID_X)$.

Obviously the third phase of key establishment can be performed in the *i*-th bootstrapping period, only if at least one of the nodes belongs to the *i*-th generation. Otherwise if both nodes have deleted the key K_i the protocol of Zhu et al., (2003) cannot be executed remotely. This however is not a strong assumption since an incoming node will initiate all the required remote key establishment during the *i*-th bootstrapping period.

4 SECURITY OF THE MULTI-LAYER SCHEME

4.1 Security of the 1st phase

By using the hybrid protocol of Kotzanikolaou et al., (2005a) the clusterhead nodes perform authenticated key establishment with explicit key confirmation. Authentication is based on Implicit Certificates and key confirmation is based on each node's proving the knowledge of the corresponding Elliptic Curve secret key. Furthermore, the use of the hybrid protocol preserves forward secrecy for the pairwise keys and also prevents impersonation and fake generation attacks from compromised clusterhead nodes of the same generation, since each node uses a unique key pair and Implicit Certificate.

4.2 Security of the 2nd phase

This phase of the local key establishment is based on the symmetric key establishment protocol of Zhu et al., (2003). Thus it inherits its security properties. Recall that the symmetric protocol of Zhu et al., (2003) performs node authentication during the key establishment. This protocol cannot prevent the fake generation attacks or impersonation attacks from compromised nodes of the same generation. However, each clusterhead will establish a key with each plain node inside its cluster only during the execution of this phase. Thus, these attacks could be applied only during this phase and only against the plain nodes. In general, the security of the first two phases relies on the security of the protocols used in these phases.

4.3 Security of the 3rd phase

The 3rd phase of key establishment combines several security features of the former two phases and thus it security must be examined in more detail. Since the key exchange is performed through intermediate nodes, several security properties must be examined.

1. **Point-to-Point Authentication.** The messages exchanged during this phase are based on the security associations that have already been established in the previous two phases. The messages that must be exchanged in order to perform the key establishment between the distant nodes are authenticated through the intermediate nodes. These nodes use the already established pairwise keys between local nodes and clusterheads and each link is authenticated by the participating nodes. Thus, this phase is not vulnerable to impersonation and fake generation attacks caused by malicious intermediate nodes and the protocol is resistant to insider attacks, such as malicious node or corrupted node attacks.
2. **End-to-end Authentication.** The end nodes are authenticated in an end-to-end manner by following the authentication protocol of Zhu et al., (2003). Indeed, the nodes will prove their identity based on the node identifier and the generation key K_i.
3. **Point-to-Point Encryption.** During the remote key establishment, the messages that are relayed between the end nodes are encrypted and decrypted in a point-to-point manner, by using the keys that were established between the nodes in the previous phases. This makes impossible for an outsider to trace and link the messages that must be exchanged between a pair of distant nodes in order to establish their pairwise key and in this way the key establishment is protected from outsiders. Note that the point-to-point encryption is only used during the exchange of the messages required for a remote key establishment. Then after the remote

nodes have established a pairwise key, they can encrypt their communication in an end-to-end manner.

4. **End-to-End Encryption**. The remote key establishment between nodes is secure against intermediate nodes that attempt to construct the paiwise key established between the end nodes. Note that the pairwise key is not exchanged between the end nodes but it is established by executing the protocol of Zhu et al., (2003). Thus an insider will not be able to generate the pairwise key, provided that the protocol of Zhu et al., (2003) is secure. Then, after the key establishment the end nodes will use end-to-end encryption in order to protect their actual communication from both insider and outsider eavesdroppers.

5 SYSTEM MODEL – PERFORMANCE ANALYSIS

In this section, the architecture of the network used to simulate and evaluate the multi-layer key establishment protocol for sensor networks is presented. We depict the topology of the simulated sensor network, the characteristics that each sensor node has, how the key establishment phase takes place and we conclude this section by presenting the simulation results and by making remarks regarding the efficiency of this multi-layer key establishment protocol for use in sensor networks.

In the simulation scenario the sensor nodes are deployed in square regions, like in Figure 2. In each square region there are randomly placed eight plain sensor nodes and one clusterhead at the center. The factory sensor nodes have the ability to transmit to distances of up to 100m or 150m, but since energy conservation is a key issue to such networks, which have very limited energy resources, the transmission range of the plain sensor nodes is set to $R_N = 5$m, whereas the transmission range of the clusterheads is set to $R_N = 9$m, to achieve communication of one clusterhead with all its neighboring ones. Therefore, the area that a plain sensor node covers is $\pi \cdot R_N^2 = 78.5$m, whereas a clusterhead sensor node covers an area of 254.34m.

The proposed scheme is evaluated through the simulation platform Network Simulator – NS-2 (www.isi.edu/nsnam/ns/). The MAC protocol is IEEE 802.11 with DCF (Distributed Coordination Function). The interface queue used is a FIFO (First In - First Out) one where the routing protocol's packets have higher priority than data packets. The maximum number of packets that this queue can hold for every node is set to 50 packets. Every node uses an omni-directional antenna. The transmission model used is the FreeSpace model, where communication takes place through a line of sight (LoS) path between the sender and the receiver. The received power in distance *d* is given by the following equation:

$$P_r(d) = \frac{P_t \cdot h_t \cdot h_r \cdot \lambda^2}{(4 \cdot \pi)^2 \cdot d^2 \cdot L} \quad (1)$$

where $\lambda = \frac{c}{f} = \frac{3 \cdot 10^8}{9.14 \cdot 10^8} = 0.32823$.

In equation (1), h_t and h_r are the heights of the antennas at the sender and the receiver, respectively. Variable P_t is the power of the transmitted signal; G_t and G_r are the gains of the antennas at the sender and the receiver, respectively, and $L(L \geq 1)$ represents the losses of the system. In the simulation: $G_t = G_r = L = 1$, $h_t = h_r = 1.5$, $P_t = 0.281838$ and $d = 5$ m or 9m.

We run several scenarios to evaluate the hybrid key establishment protocol described in (Kotzanikolaou et al., 2005a). One scenario is to estimate the time needed for two nodes to establish a key in real network conditions; a second scenario is to evaluate the required time for a sensor node to establish keys with eight neighboring nodes and finally a third scenario is run to evaluate the hybrid key establishment protocol when more than one key establishment initiations take place. All these scenarios were run using two different routing protocols, the AODV (Ad-hoc On Demand Distance Vector) and the DSDV (Destination Sequenced Distance Vector). In Table 1 below, the theoretical time for a key establishment and the results of the deployment of these three simulation scenarios are presented, while Figure 4 shows the results of the simulated times for the three scenarios.

	Virtual time ***(1-1) (sec)***	***Real time*** ***(1-1) (sec)***	***Real time*** ***(1-8) (sec)***	***Real time*** ***(4-8) (sec)***
AODV	0.678	0.6994	4.7499	6.0249
DSDV	0.678	0.6975	4.6984	4.9404

Table 1: *The results of the theoretical key establishment and the three simulation scenarios*

The multi-layer key establishment protocol was evaluated by creating a similar agent in the simulation platform, which was attached to all the sensor nodes in the network. This agent is responsible for conducting the three phases described in Section 3 as follows:

Firstly, the wireless sensor nodes are deployed inside the sixteen clusters, as described above, *i.e.* eight plain sensor nodes are randomly placed in each cluster and one clusterhead node at the center of each cluster. In the beginning of the simulation, the creation of the routing tables of each sensor node takes place when the DSDV routing protocol is used.

After the completion of this phase, starts the first phase of the multi-layer key establishment protocol. Each clusterhead establishes a key with all its eight neighbouring clusterheads, using the hybrid protocol described in (Kotzanikolaou et al., 2005a). In our simulation scenario, only the four clusterheads in the central square regions initiated the key establishment protocol with their neighboring clusterheads. During this phase, the plain sensor nodes remain silent, for example, by switching to a power-saving mode. In the second phase, the key establishment protocol inside the clusters is initiated. Each plain sensor node will perform a key establishment with the clusterhead of the cluster where it belongs. When this phase ends, each sensor node has the option of establishing a key with another node that belongs to a different cluster.

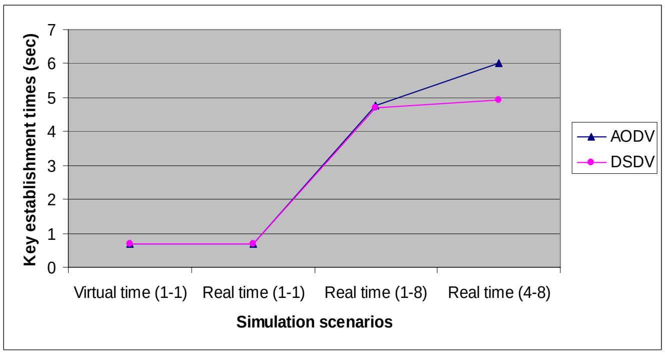

Figure 4 *The simulation results for the various scenarios*

In this last phase, we simulated a scenario where a plain sensor node firstly establishes a key with another plain node in a neighboring cluster, secondly with a plain node two clusters away and thirdly with a plain node three clusters away. All the steps of the key establishment protocol go through the intermediate clusterheads, as shown in figure 3. The theoretical times for each phase and the simulation times are presented in Table 2.

	Hybrid (Between clusterheads)	Symmetric (Inside the clusters)	1 cluster away	2 clusters away	3 clusters away
Virtual time (sec)	< 5.424 *	0.025	0.088	0.102	0.136
Real time (sec)	4.9404	1.5787	0.0889	0.1252	0.3784

** depends on several factors*

Table 2: *The results of the theoretical key establishment and the three simulation scenarios*

Figure 5 presents the total real (simulation) time required for a key establishment for a remote key establishment for nodes of various distances. The resulting times include the costs derived from all the three phases of the protocol.

During the execution of the hybrid key establishment protocol between the neighbouring clusterheads, only the clusterheads located in the central clusters initiate a key establishment. This results in having different number of key establishments taking place per node, since some key establishments will have already been completed in one direction. Therefore, the time required for this phase is significantly reduced. The symmetric key establishment time inside the clusters is quite larger compared to the theoretical time required for this phase. This is attributed to the fact that in our simulation scenario we consider the worst case, where the plain sensor nodes initiate the symmetric key establishment simultaneously resulting in having too many collisions and retransmissions after a random time defined by the backoff algorithm. This time can be improved by having the clusterhead initiate the symmetric key establishment and not the other way around. Comparing the simulation times of the third scenario with the theoretical times in Table 2, we observe that when the number of clusterheads, which the packets of the key establishment protocol between two plain sensor nodes have to traverse, increases then the difference between the simulated time and the theoretical time increases as well. This happens because the packets have to be transmitted passing through several sensor nodes and might be delayed if there are other nodes that may have ongoing transmissions, even if these transmissions are of the routing protocol.

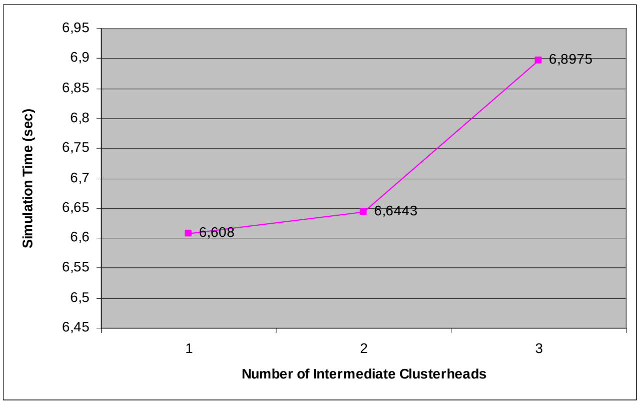

Figure 5 *Total simulation time for remote key establishments of nodes of various distances*

6 SYSTEM MODEL – PERFORMANCE ANALYSIS

Distributed Sensor Networks (DSN) are becoming increasingly popular as they can be used in a variety of civil applications, or in secure-critical domains such as military applications: hundreds or even thousands of such nodes could be used to collect information in combat scenarios such as tracking and reporting hostile troop movements, or detecting chemical and biological weapons. In such applications, *multi-phase* deployment may be a requirement, *i.e.* the network is updated with other nodes in future time periods, in order to extend the network or replace erroneous nodes.

Depending on the environment where nodes are deployed, appropriate protection measures should be taken for data confidentiality, integrity and authentication within the sensor network. An interesting research area is key establishment in self-organising networks (Eschenauer and Gligor, 2004), *i.e.* networks that do not rely on any fixed infrastructure. In such networks key establishment techniques must be very efficient. For this reason, several symmetric protocols for self-organising DSNs have been developed (Dutertre et al., 2004; Zhu et al., 2003), which are not appropriate for highly sensitive applications, since they are vulnerable to impersonation and fake generation attacks. Recent hybrid key establishment protocols (Huan et al., 2003; Kotzanikolaou et al. 2005a) seem to be more resilient to impersonation attacks than the symmetric ones as they combine symmetric techniques with limited public-key cryptography. Furthermore, they allow multiphase key. Unfortunately, although these hybrid key establishment protocols are efficient for DSNs, they are still more expensive than the symmetric protocols.

In this paper, we propose a multi-layer key establishment scheme, which combines hybrid and symmetric key establishment techniques. The nodes are clustered in geographic areas: In each cluster there exists one node (the clusterhead node) with extended capabilities. Each clusterhead uses hybrid key establishment of Kotzanikolaou et al., (2005a) to exchange pairwise keys with other neighbouring clusterheads. Furthermore, plain nodes belonging in the same cluster use symmetric key establishment (Zhu et al., 2003) in order to exchange pairwise keys. Finally, plain nodes belonging to different clusters employ the intermediate clusterheads to establish keys using the symmetric technique. In comparison with the hybrid protocol (Kotzanikolaou et al., 2005b) simulation results of the multi-layer key establishment protocol show a reasonable increase in performance, which is reasonable since the hybrid protocol is not extensively used.

Each phase of the key establishment scheme creates a key establishment layer. The first phase generates a key establishment layer between the clusterheads. The use of the hybrid protocol allows for secure multi-phase deployment of sensor nodes and prevents impersonation attacks from compromised clusterhead nodes of the same generation. The second phase generates a key establishment layer inside each cluster. It is based on a symmetric key establishment protocol (Zhu et al., 2003) and thus it inherits its security properties. Finally, the third phase uses the two former layers in order to generate a key establishment layer between distant plain nodes. This phase is not vulnerable to impersonation and fake generation attacks since all messages exchanged between the distant nodes are encrypted and authenticated through the intermediate clusterheads.

Sensor networks will be massively deployed only when security and efficiency research challenges are addressed. Further research and implementation results into energy efficient cryptographic primitives are necessary. Our findings show that it can be feasible to use limited public key cryptography as a supplementary security primitive, for special uses of low-energy computing devices.

REFERENCES

Certicom Research, (2000) 'Standard for efficient cryptography', SEC 1: EC Cryptography. Ver. 1.0.

Dutertre, B., Cheung, S., and Levy, J. (2004) 'Lightweight Key Management in Wireless Sensor Networks by Leveraging Initial Trust', *TR SRI-SDL-04-02*, April.

Eschenauer, L., and Gligor, V. D. (2004) 'A key-management scheme for distributed sensor networks', *Proc. of 9th CCS ACM conference*.

Gaubatz, G., Kaps, J.P. and Sunar, B. (2004) 'Public key cryptography in sensor networks – revisited', *Proceedings of 1st ESAS Conference*.

Huang, Q, Cukier, J., Kobayashi, H., Liu, B. and Zhang, J (2003) 'Fast authenticated key establishment protocols for self-organizing sensor networks', *Proceedings of 2nd ACM WSNA Conference*, pp. 141–150.

Kotzanikolaou, P., Magkos, E., Douligeris, C., and Chrissikopoulos, V. (2005) 'Hybrid Key Establishment for Multiphase Self-Organized Sensor Networks', *Proceedings of the 1st IEEE International Workshop on Trust, Security and Privacy for Ubiquitous Computing*, pp. 581 –587.

Kotzanikolaou, P., Vergados, D.D., and Stergiou, G. (2005) 'Performance Analysis of a Hybrid Key Establishment Protocol for Wireless Sensor Networks', *Proceedings of the IEEE International Symposium on Multimedia (ISM 2005),* pp.719-724.

Malan, D. Welsh, M. and Smith, M. (2003) 'A Public-Key Infrastructure for Key Distribution in TinyOS Based on Elliptic Curve Cryptography', *Proceedings of 1st SACN Conference,* IEEE.

Schnorr C. (1991), 'Efficient Signature Generation by Smart Cards', *Journal of Cryptology*. Vol. 4, pp. 161–174.

Zhu, S., Setia, S. and Jajodia, S. (2003) 'LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks', in: *Proceedings of ACM CCS'03,* Washington D.C., October 2003.

WEBSITES

The Network Simulator – NS-2, at http://www.isi.edu/nsnam/ns/