On Communication Security in Wireless Ad-Hoc Sensor Networks

Sasha Slijepcevic,
Miodrag Potkonjak
Computer Science Department, UCLA
{sascha,miodrag}@cs.ucla.edu

Vlasios Tsiatsis, Scott Zimbeck,
Mani B. Srivastava
Electrical Engineering Department, UCLA
{tsiatsis, szimbeck, mbs}@ee.ucla.edu

Abstract

Networks of wireless microsensors for monitoring physical environments have emerged as an important new application area for wireless technology. Key attributes of these new types of networked systems are the severely constrained computational and energy resources, and an ad hoc operational environment. This paper is a study of the communication security aspects of these networks. Resource limitations and specific architecture of sensor networks call for customized security mechanisms. Our approach is to classify the types of data existing in sensor networks, and identify possible communication security threats according to that classification. We propose a communication security scheme where for each type of data we define a corresponding security mechanism. By employing this multitiered security architecture where each mechanism has different resource requirements, we allow for efficient resource management, which is essential for wireless sensor networks.

Keywords—wireless, sensor, networks, communication

1. Introduction

Wireless sensor networks, applied to monitoring physical environments, have recently emerged as an important application resulting from the fusion of wireless communications and embedded computing technologies [1][3][13][18][19].

Sensor networks consist of hundred or thousands of sensor nodes, low power devices equipped with one or more sensors. Besides sensors, a sensor node typically contains signal processing circuits, microcontrollers, and a wireless transmitter/receiver. By feeding information about the physical world into the existing information infrastructure, these networks are expected to lead to a future where computing is closely coupled with the physical world and is even used to affect the physical world via actuators. Potential applications include monitoring remote or inhospitable locations, target tracking in battlefields, disaster relief networks, early fire detection in forests, and environmental monitoring.

While recent research has focused on energy efficiency [14], network protocols [6], and distributed databases, there is much less attention given to security. The only work that we are aware of is [11]. However, in many applications the security aspects are as important as performance and low energy consumption. Besides the battlefield applications, security is critical in premise security and surveillance, and in sensors in critical systems such as airports, hospitals, etc. Sensor networks have distinctive features, the most important ones being constrained energy and computational resources. To accommodate those differences existing security mechanisms must be adapted or new ones created.

The main contributions of our work are:

- An assessment of communication security threats in sensor networks.
- Separate security mechanisms for data with various sensitivity levels. Such separation allows efficient resource management that is essential for wireless sensor networks.
- A location-based scheme that protects the rest of a network, even when parts of the network are compromised.

Our approach to communication security in sensor networks is based on a principle stated in [12], that says that data items must be protected to a degree consistent with their value. In the particular architecture [4], for which we are developing our communication security scheme, we differentiate between three types of data sent through the network:

1. Mobile code
2. Locations of sensor nodes
3. Application specific data

Following this categorization, we specify the main security threats and the appropriate security mechanisms:

Proceedings of the Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02)
1080-1383/02 $17.00 © 2002 **IEEE**

- Fabricated and malicious mobile code injected into a network can change the behavior of the network in unpredictable ways.
- Acquiring locations of sensor nodes may help an adversary to discover locations of sensor nodes easier than using radio location techniques.
- Protection of application specific data depends on the security requirements of a particular application. In a target tracking application, which was a test case for the given security scheme, we treated the application specific data as the least sensitive type of data.

Our main goal is to minimize security related energy consumption. By offering a range of security levels we ensure that the scarce resources of sensor nodes are used accordingly to required protection levels. There are many other important issues for security in sensor networks, e.g. physical protection of the sensitive data in sensor nodes, and the system-level security. However, those topics are outside of the scope of this paper. The complexity of building tamper-proof circuits that could protect sensitive information held in a node is described in [2].

In Section 2, we describe the SensorWare network architecture for which the communication security scheme is developed. Section 3 categorizes possible threats to a sensor network. In Section 4, we propose the communication security mechanisms corresponding to the defined types of data. Section 5 describes the implementation environment. Section 6 discusses related work, while Section 7 concludes the paper.

2. Sensor Network Architecture

In this section, we briefly describe the SensorWare network architecture based on the research at UCLA and Rockwell Science Center [16]. We point out the aspects of the architecture that impact the design of the security scheme. The most important elements of the architecture are: localized algorithms, local broadcast model of communication, and mobile code.

2.1. Localized Algorithms

The most distinctive feature of sensor networks is the limited energy available to sensor nodes. Consequently, careful budgeting of the available energy becomes a fundamental design principle. Keeping in mind that communication between nodes consumes a significant amount of the energy resources, applications and system software are expected to achieve a required level of performance while minimizing the amount of traffic in the network. In the SensorWare architecture, the applications are designed based on localized algorithms, where nodes triggered by an event exchange messages within an immediate neighborhood. Only one node aggregates all the sensor readings and sends the combined data to a gateway node, which is one of the sensor nodes in a network capable of serving as a proxy between the network and the user.

2.2. Local broadcast

In sensor networks, local broadcast is a fundamental communication primitive. Local broadcast is necessary to build and maintain sensor networks architectures, and to support the exchange of the data about detected events. Any node in the network can be a sender or a receiver of a broadcast message. These properties of sensor networks have a significant impact on the security. In our security scheme, we use shared symmetric keys for encryption. Such a solution simplifies the key management and retains the energy efficiency of local broadcast, but does not offer strong authentication.

2.3. Code Mobility

The code mobility paradigm is essential in sensor networks for two reasons:

1. Limited storage available to nodes does not allow keeping all application on a node at all times.
2. Applications that a network should run may not be known at the time of deployment of the network.

Since manual reconfiguration of sensor nodes after deployment is not feasible, the support for mobile code is additionally important.

3. Security Threats

Wireless networks, in general, are more vulnerable to security attacks than wired networks, due to the broadcast nature of the transmission medium. Furthermore, wireless sensor networks have an additional vulnerability because nodes are often placed in a hostile or dangerous environment where they are not physically protected.

To demonstrate, on an example, some of the security threats and our corresponding protection mechanisms, we simulated and implemented a target tracking application. The nodes that detect a target in an area exchange messages containing a timestamp, the location of the sending node and other application-specific information. When one of the nodes acquires a certain number of messages such that the location of the target can be approximately determined, the node sends the location of the target to the user.

Not only the application messages are exchanged through the network, but also mobile code is sent from

node to node. Because the security of mobile code greatly affects the security of the network, we consider protection of the messages containing mobile code as an important part of our communication security scheme.

For the types of data specified in Section 1, we list the possible threats to a network if communication security is compromised:

1. Insertion of malicious code is the most dangerous attack that can occur. Malicious code injected in the network could spread to all nodes, potentially destroying the whole network, or even worse, taking over the network on behalf of an adversary. A seized sensor network can either send false observations about the environment to a legitimate user or send observations about the monitored area to a malicious user.

2. Interception of the messages containing the physical locations of sensor nodes allows an attacker to locate the nodes and destroy them. The significance of hiding the location information from an attacker lies in the fact that the sensor nodes have small dimensions and their location cannot be trivially traced. Thus, it is important to hide the locations of the nodes. In the case of static nodes, the location information does not age and must be protected through the lifetime of the network.

3. Besides the locations of sensor nodes, an adversary can observe the application specific content of messages including message IDs, timestamps and other fields. Confidentiality of those fields in our example application is less important then confidentiality of location information, because the application specific data does not contain sensitive information, and the lifetime of such data is significantly shorter.

4. An adversary can inject false messages that give incorrect information about the environment to the user. Such messages also consume the scarce energy resources of the nodes. This type of attack is called *sleep deprivation torture* in [17].

4. Communication Security Scheme

After we defined the three types of data in the SensorWare network, and the possible threats to the network, in this section we define the elements of the security scheme. The three security levels described here are based on private key cryptography utilizing group keys. Applications and system software access the security API as a part of the middleware defined by the SensorWare architecture. Since all three types of data contain more or less confidential information, the content of all messages in the network is encrypted.

We assume that all sensor nodes in the network are allowed to access the content of any message. As we said before, we only deal with communication security. Protection of data within a node is not discussed here.

The deployment of security mechanisms in a sensor network creates additional overhead. Not only does latency increases due to the execution of the security related procedures, but also the consumed energy directly decreases the lifetime of the network. To minimize the security related costs we propose that the security overhead, and consequently the energy consumption, should correspond to sensitivity of the encrypted information. Following the taxonomy of the types of data in the network, we define three security levels:

- Security level I is reserved for mobile code, the most sensitive information sent through the network,
- Security level II is dedicated to the location information conveyed in messages,
- The security level III mechanism is applied to the application specific information.

The strength of the encryption for each of security levels corresponds to the sensitivity of the encrypted information. Therefore, the encryption applied at level I is stronger than the encryption applied at level II, while the encryption on level II is stronger than the one applied at level III.

Different security levels are implemented either by using various algorithms or by using the same algorithm with adjustable parameters that change its strength and corresponding computational overhead. Using one algorithm with adjustable parameters has the advantage of occupying less memory space.

We selected RC6 [15]. RC6 is suitable for modification of its security strength because it has an adjustable parameter (number of rounds) that directly affects its strength. The overhead for the RC6 encryption algorithm increases with the strength of the encryption measured by the number of rounds [10]. Our implementation results presented in Section 5 also demonstrate that property.

The multicast model of communication inherent for the SensorWare architecture suggests deployment of group keys. Otherwise, if each pair of nodes would require a key or a pair of keys, communication between the nodes would have to be unicast based. This would significantly increase the number of messages. Since the addition of security in a sensor network must not require the change of the whole sensor network architecture, group keys are utilized.

All nodes in the network share an initial set of master keys. The number of the keys depends on the estimated lifetime of the network. The longer the lifetime, the more keys are needed in order to expose less material for a "known ciphertext" attack. The alternative approach where the keys would be established dynamically and propagated through the network is not acceptable. It would require such a protocol that guarantees that all nodes received a key. Such a requirement is not feasible in

a network where the nodes do not keep track of their neighbors.

One of the keys from the list of master keys is active at any moment. The algorithm for the selection of a particular key is based on a pseudorandom generator running at each node with the same seed. Periodically and synchronously on each node, a new random number is generated and used to provide and index to an entry in the table of the available master keys. This entry contains the active master key. The keys for three levels of security corresponding to the three types of data are then derived from the active master key.

4.1. Security Level I

The messages that contain mobile code are less frequent than the messages that the application instances on different nodes exchange. It allows us to use a strong encryption in spite of the resulting overhead. For information protected at this security level, nodes use the current master key. The set of master keys, the corresponding pseudorandom number generator, and a seed are credentials that a potential user must have in order to access the network. Once when the user obtains those credentials, she can insert any code into the network. If a malicious user breaks the encryption on this level using a "brute force" attack, she can insert harmful code into the network.

4.2. Security Level II

For data that contains locations of sensor nodes, we provide a novel security mechanism that isolates parts of the network, so that breach of security in one part of the network does not affect the rest of the network.

According to our assumptions about the applications expected to run in sensor networks, the locations of sensor nodes are likely to be included in the majority of messages. Thus, the overhead that corresponds to the encryption of the location information significantly influences the overall security overhead in the network. This must be taken into account when the strength of the encryption at this level is determined. Since the protection level is lower for the location information than for mobile code, the probability that the key for the level II can be broken is higher. Having the key, an adversary could potentially locate all nodes in the network. To constrain the damage to only one part of the network, we propose the following security mechanism. Sensor nodes use location-based keys for level II encryption. The location-based keys enable separation between the regions where the location of nodes are compromised and the areas where nodes continue to operate safely.

The area covered by a sensor network is divided into cells. Nodes within one cell share a common location-based key, which is a function of a fixed location in the cell and the current master key. Between the cells, there is a bordering region whose width is equal to the transmission range. Nodes belonging to those regions have the keys for all adjacent cells. This ensures that two nodes within a transmission range from each other have a common key. The dimensions of the cells must be big enough so that the localized nature of the algorithms in the network ensures that the traffic among the cells is relatively low, compared to overall traffic. The areas can be of an arbitrary shape with the only requirement that the whole sensor terrain is covered. A division of the area in uniformly sized cells is the most appropriate solution, because it allows a fast and easy way for a node to determine its cell membership. We divide the network into hexagonal cells, since it ensures that the gateway nodes have at most three keys.

A part of the bootstrapping mechanism for sensor nodes is the process of determining their cell membership. In that process, we use the notion of *extended cell*. An extended cell is a hexagonal cell, which has the same center as the original cell and the distance between its sides and the sides of the original cell is equal to the transmission range of the sensor nodes. The extended cell contains the original cell and corresponding bordering regions. Fig. 1 shows three neighboring cells and their corresponding extended cells. Each node compares its location against each extended cell and determines if it is in an extended cell or not. If a node is within the extended cell of C_x, it will have the key of C_x, K_{Cx}. The nodes within the bordering regions (shaded areas) have multiple keys as shown. For example, the nodes that are adjacent to cells C_1 and C_2 have two keys: K_{C1} and K_{C2}, respectively.

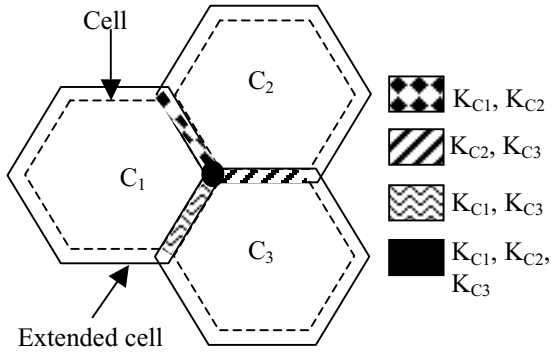

Figure 1. Cells, Extended cells and areas with multiple keys

4.3. Security Level III

We encrypt the application specific data using a weaker encryption than the one used for the two aforementioned types of data. The weaker encryption requires lower computational overhead for application specific data. Additionally, the high frequency of messages with application specific data prevents using stronger and resource consuming encryption. Therefore, we apply an encryption algorithm that demands less computational resources with a corresponding decrease in the strength of security.

The key used for the encryption of the level III information is derived from the current master key. The MD5 hash function accepts the master key and generates a key for level III. Since the master key is periodically changed, the corresponding key at this level follows those changes.

In the discussion above the major assumptions of the all the proposed security schemes is that the sensor nodes are perfectly time synchronized and have exact knowledge of their location. It is not unrealistic [5] that the nodes can be synchronized up to μs.

5. Implementation

As a part of a proof of concept implementation, we ported the encryption routines of RC6 on the Rockwell WINS sensor nodes. Each operates with an Intel StrongARM 1100 processor running at 133 MHz, 128KB SRAM, 1MB Flash Memory, a Conexant DCT RDSSS9M radio, a Mark IV geophone and RS232 external interface. The radios transmit at 100Kbps with the transmission power of 1mW, 10mW, or 100mW. Using the ARM System Developers Kit profiling tools, we measured the clock cycles spend for encryption and decryption of a single 128 bit block with a key of length 128, versus the number of algorithmic rounds. In the AES candidate report [10] the number of rounds, determines the security strength of an algorithm. In this report for each algorithm a minimum number of rounds for which the algorithm is considered to be secure ($R_{\min}$) is presented.

Based on this quantity, the *security margin* of an encryption algorithm is defined as the percentage of deviation of the actual number of rounds from $R_{\min}$:

$$M_s = \frac{R - R_{\min}}{R_{\min}}.$$

Fig. 2 depicts the total clock cycles for encryption and decryption of a single 128-bit block with a 128-bit key versus the number of rounds.

As the figure shows, there is a linear relationship between the clock cycles and the number of rounds. As also shown from the equation above, increasing the number of rounds, increases the security margin but the overhead for each block is also increased.

The specification of the Rockwell WINS node can be found in [9] and [20]. The maximum energy saving is achieved when the radio transmission power is set to 1mW. To send a block of 128 bits, the radio consumes 1.28 μJ. The processor consumes 3.9 μJ to encrypt the block using 32 rounds, which corresponds to security level I. The energy consumed when the same block is encrypted using 22 rounds, which corresponds to level III, is 2.7 μJ. Therefore, if a message contains the data that is encrypted on security level III the energy consumption decreases by 23% compared to a scheme where all data is encrypted on level I. For the transmission power of 10mW, the maximum savings are only 2%. It is important to mention that the messages containing the location and the application specific data are likely to occur much more frequently than the messages containing mobile code, for which the consumed energy is the same for the multitiered scheme and the scheme with only one encryption level.

6. Related Work

The issue of security in wireless sensor networks has not attracted much attention. The only work in that area known to us is [11]. The sensor network architecture discussed there significantly differs from the SensorWare. In [11], the sensor network relies on the existing infrastructure of the energy unconstrained base stations that communicate with the resource constrained nodes. The security protocol μTESLA, built for such an environment, mainly supports the authenticated broadcast

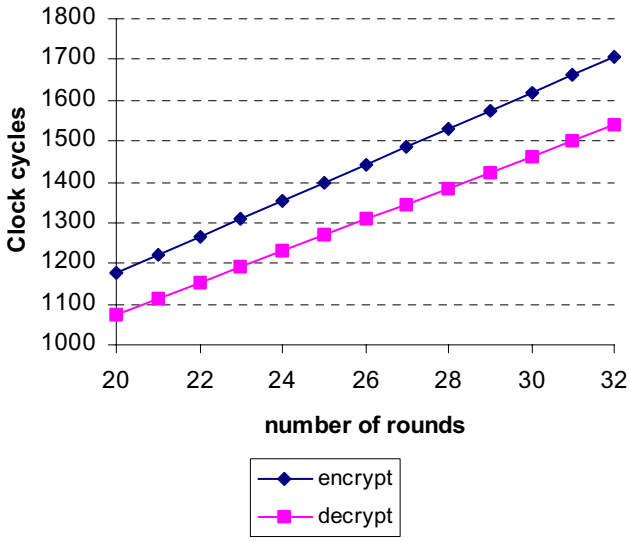

Figure 2. Encryption and decryption clock cycles versus the number of rounds for RC6

from a base station to surrounding nodes. Even if a node has to send a broadcast message, it must rely on support from a base station. The protocol ensures authentication of broadcast messages by distributing a key after the messages encrypted with that key. Base stations are part of a trusted computing base, and it is assumed that they cannot be compromised. In our architecture all nodes can be senders and receivers of broadcast messages. In order to achieve a strong authentication offered by μTESLA in our architecture, each node would have to have its own key known to all other nodes in the network. In a network with possibly thousands of nodes, such a solution does not scale well.

In secure multicast for wired and mobile networks [7][8] the main problem is key management, i.e. the re-keying overhead when users join and leave the group. In sensor networks the problem is different, since the sensor nodes do not leave the group, and newly deployed nodes are not forbidden to access the messages generated before their deployment. The goal in sensor networks is to keep external adversaries out of the group in an energy and computationally efficient way. However, the approach of dividing a group into subgroups and having gateways for the inter-subgroup communication, used in secure multicast, is similar to our approach of the division of the sensor terrain in location based key areas.

7. Conclusion

In this paper, we propose a communication security scheme for sensor networks. The straightforward approach to the secure communication in sensor networks could be the application of a single security mechanism for all data in the network. However, if the mechanism is chosen according to the most sensitive data in the network, security related resource consumption might be unacceptable. On the other hand, a less consuming mechanism could allow for serious security threats. Therefore, the solution lies in the identification of appropriate security requirements for various types of data and the application of suitable security mechanisms. Using the target tracking application as an example, and the SensorWare architecture as a target platform, we define here some security challenges in sensor networks, identify different types of data, and propose and implement elements of a communication security scheme.

Secure communication, which is the topic of this paper, is only one of the security issues in sensor networks. An important security concern in the SensorWare architecture is the deployment of mobile code. Besides sensor networks, there are other systems, where flexibility is required, but the security of a system must not be jeopardized (Java Virtual Machines in Web browsers is one of the well known examples).

The research described in this paper was founded in part by DARPA's SensIT program under AFRL Contract F30602-99-1-0529. The views expressed in this paper are those of the authors, and do not necessarily represent those of DARPA or AFRL.

References

[1] H. Abelson et. al., "Amorphous Computing", Communication of ACM, vol.43, vol. 5, May 2000, pp. 74-82.

[2] R. Anderson, M. Kuhn, "Tamper resistance—a Cautionary Note", In Proceedings of the Second USENIX Workshop on Electronic Commerce, 1996.

[3] G. Borriello, R. Want, "Embedding the Internet: Embedded Computation Meets the World Wide Web", Communication of ACM, vol.43, no.5, May 2000, pp. 59-66.

[4] DARPA SensIT program. http://dtsn.darpa.mil/ixo/sensit.asp

[5] J. Elson, D. Estrin, "Time Synchronization for Wireless Sensor Networks", In Proceedings of the 2001 IPDPS, Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[6] D. Estrin, R. Govindan, J. Heidemann, "Embedding the Internet: Introduction", Communications of the ACM, vol.43, no.5, May 2000, pp. 38-41.

[7] L. Gong, N. Shacham, "Multicast Security and its Extension to a Mobile Environment", Wireless Networks, vol.1, (no.3), 1995, pp. 281-295.

[8] P. Kruus, J. Macker, "Techniques and Issues in Multicast Security", MILCOM 98, vol.3, Boston, MA, USA, 1998, pp. 1028-32.

[9] J. Agre, L. Clare, G. Pottie, N. Romanov, "Development Platform for Self-Organizing Wireless Sensor Networks", Proceedings of SPIE AeroSense'99 Conference on Digital Wireless Communication, Orlando, FL, USA, April 1999.

[10] J. Nechvatal, E. Barker, D. Dodson, M. Dworkin, J. Foti, E. Roback, "Status Report on the First Round of the Development of the Advanced Encryption Standard", http://csrc.nist.gov/encryption/aes/round1/r1report.htm.

[11] A. Perrig, R. Szewczyk, V. Wen, D. Culler, J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", MOBICOM 2001, Rome, Italy, June 2001.

[12] C. P. Pfleeger, "Security in Computing", Second Edition, Prentice Hall, 1997.

[13] G. J. Pottie, W. J. Kaiser, "Embedding the Internet: Wireless Integrated Network Sensors", Communications of ACM, vol.43, no.5, May 2000, pp.51-58.

[14] J. Rabaey, J. Ammer, J. L. da Silva, D. Patel, "PicoRadio: Ad-hoc Wireless Networking of Ubiquitous Low-Energy Sensor/Monitor Nodes", Workshop on VLSI, April 2000.

[15] R. L. Rivest, M.J.B. Robshaw, R. Sidney, and Y.L. Yin, "The RC6 Block Cipher", AES submission, Jun 1998. http://theory.lcs.mit.edu/~rivest/rc6.pdf.

[16] SensorWare Architecture http://www.rsc.rockwell.com/wireless_systems/sensorware/

[17] F. Stajano, R. Anderson, "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks", 3rd AT&T Software Symposium, Middletown, NJ, October 1999.

[18] G. S. Sukhatme, M. J. Mataric, "Embedding the Internet: Embedding Robots into the Internet", Communication of ACM, vol.43, vol.5, May 2000, pp.67-73.

[19] D. Tennenhouse, "Embedding the Internet: Proactive Computing", Comm. of ACM vol.43, no.5, May 2000, pp. 43-50.

[20] V. Raghunathan, C. Schurgers, S. Park, M. B. Srivastava, "Energy-aware wireless microsensor networks", IEEE Signal Processing Magazine, vol.19, (no.2), IEEE, March 2002. pp. 40-50.