North Dakota State University--Fargo

From the SelectedWorks of Jeremy Straub

April, 2017

An Energy-Efficient and Secure Hybrid Algorithm for Wireless Sensor Networks Using a Mobile Data Collector

Karanam R Dayananda, North Dakota State University--Fargo
Dr. Jeremy Straub, North Dakota State University--Fargo

Available at: https://works.bepress.com/jeremy_straub/350/
An Energy-Efficient and Secure Hybrid Algorithm for Wireless Sensor Networks Using a Mobile Data Collector
Karanam Ravichandran Dayananda, Jeremy Straub
Dept. of Computer Science, North Dakota State University, 1320 Albrecht Blvd, Room 258, Fargo, ND, USA 58102

ABSTRACT
This paper proposes a new hybrid algorithm for security, which incorporates both distributed and hierarchal approaches. It uses a mobile data collector (MDC) to collect information in order to save energy of sensor nodes in a wireless sensor network (WSN) as, in most networks, these sensor nodes have limited energy. Wireless sensor networks are prone to security problems because, among other things, it is possible to use a rogue sensor node to eavesdrop on or alter the information being transmitted.

To prevent this, this paper introduces a security algorithm for MDC-based WSNs. A key use of this algorithm is to protect the confidentiality of the information sent by the sensor nodes. The sensor nodes are deployed in a random fashion and form group structures called clusters. Each cluster has a cluster head. The cluster head collects data from the other nodes using the time-division multiple access protocol. The sensor nodes send their data to the cluster head for transmission to the base station node for further processing. The MDC acts as an intermediate node between the cluster head and base station. The MDC, using its dynamic acyclic graph path, collects the data from the cluster head and sends it to base station. This approach is useful for applications including warfighting, intelligent building and medicine. To assess the proposed system, the paper presents a comparison of its performance with other approaches and algorithms that can be used for similar purposes.

Keywords: wireless sensor network, WSN, cluster head, base station, mobile data collector, security, sensor node

1. INTRODUCTION
Wireless sensor networks (WSNs) are used to collect information from a given region or location. The WSN consists of tiny devices which are equipped with microprocessors, wireless interfaces, and power sources. Each sensor node has a sensing capability and a constrained energy supply, memory, communication ability and computational power.

As the WSN’s power resources are limited and limit the WSN’s lifetime, there is a need for energy efficient protocols to improve the network lifetime and scalability. To organize sensor nodes in effective way, clusters are formed, which are used to reduce the consumption of energy, as compared to other techniques. There are several routing protocols based on the use of clustering algorithms1,2.

In traditional networks, security tends to always be an issue. However, when designing WSN applications, additional security needs must be considered. The limited resources and scalability makes security algorithms difficult to implement in a WSN. This is due to the cryptographic techniques used in many algorithm which requires complex processing, which drains energy.

One of the applications of a wireless sensor network is for military use. This includes battlefield reconnaissance surveillance and tracking: monitoring friendly forces, ammunition and equipment, and nuclear, biological, and chemical attack detection. WSNs form an integral part of military control, command, communication, computing, surveillance, and targeting systems. Self-organization, fault tolerance, and random installation makes sensor networks a promising sensing technique for military applications.

For system to be secure, security must be integrated into every component. Hence, a component without or with limited security becomes a point of attack. Establishment of key and trust relationships is required to create a secure channel. The simplest solution for establishment of a key relationship is to use a network-wide shared key. However, with this approach,
compromise of a single node would reveal the secret key allowing decryption of the entire network. One approach to solve this issue is providing a set of keys and changing nodes to use another set-member regularly.

The advantage of the symmetric cryptography is its speed. With symmetric cryptography, the encrypted data can be transferred on the link even if there is a possibility that the data will be intercepted. Since there is no key transmitted with the data, the chances of data being decrypted are reduced. A symmetric cryptosystem uses password authentication to prove the receiver’s identity.

In order to build an effective security mechanism, the code space of the security algorithm needs to be reduced as the WSN has limited capabilities in terms of power and memory. The TelosB sensor, for example has a 8 MHz RISC CPU with 4-10K RAM, 16 bit, 48 K Program memory, and 1024 K flash memory. So the software for the sensor needs to be quite small due to these restrictions. Hence, the code size of the security algorithm must be small. The operating system used in sensor nodes is TinyOS\textsuperscript{1} which is a small, event driven operating system which uses almost half of the 8Kbytes of flash memory remaining for security and application uses.

The limited energy constraint makes it impractical to use most current encryption algorithms that are designed for powerful processors. The sensor nodes are not even capable of running asymmetric cryptographic algorithms (such as RSA\textsuperscript{4} with 1024 bits). The challenge, thus, for sensor nodes is to broadcast authenticated data to the entire sensor network.

This paper discusses the challenges and proposes an algorithm for WSNs that is responsive to their limited energy and memory. With this algorithm, the network is divided into inner and outer zones. In section 6, the algorithms for the inner and outer zones are introduced. The inner zone incorporates symmetric cryptography between the cluster head and the base station. However, the outer zone uses a hybrid security technique. The member nodes and cluster head use symmetric cryptography and the mobile node uses asymmetric cryptography for encrypting their data. This is because the mobile node has resources similar to the base station. The decryption is also different in the case of the inner and outer zones. In the inner zone, encrypted data is sent to the base station. The base station decrypts it using the asymmetric key used for encrypting the data. In the outer zone, encryption takes place in three places. First, encryption takes place in the sensor node. Second, encryption takes place in the cluster head. Third, encryption takes place in the mobile node. The encryption at the sensor node and the cluster head uses symmetric key cryptography. However, the mobile node uses asymmetric key cryptography. The decryption at the base station uses asymmetric key cryptography.

2. BACKGROUND

This section reviews the technical challenges related to potential applications for and previous work surrounding WSNs. In particular, it focuses on prior work on security for WSNs.

Most wireless sensor network applications need protection against injection, eavesdropping, and the modification of data packets. The standard approach used for this security is cryptographic techniques\textsuperscript{2}. Point-to-point communications need an end-to-end cryptographic solution. However, this method seems to be incompatible with WSN’s passive participation and local broadcast methodology. Using link layer cryptography simplifies key set up and supports local broadcasting and passive participation. However, an intruder can eavesdrop on intermediate nodes and the messages can be read, altered or modified. Cryptographic techniques require extra computation capabilities and may increase the packet size. Hardware support for cryptography can increase the efficiency, but may increase the financial cost of the network.

Researchers have proposed a variety of relevant protocols. The first step required for security is the establishment of shared keys. One approach for this is public key cryptography (such as Diffie-Hellman key establishment). However, this is not suitable for WSNs. It cannot be used because of the complex processing required by the algorithm, given the WSN’s resource constraints. The advantage of this algorithm is that the node can set up for secure communications with any other node in the network. Another approach would be preconfiguring the entire network with symmetric keys for each node-to-node communication channel, though this does not scale to meet the expected requirements of larger systems. Under this model, in a sensor network installed with n sensor nodes, each node would have to store (n-1) keys in its memory.
This approach is not feasible because of the limited memory constraint of the sensor nodes. Another option would be bootstrapping the keys using a trusted base station. The sensor nodes would share a single key with the base station and communicate with other nodes through the base station. If there is failure at the base station, the network would be impaired.

Researchers developed a random-key redistribution protocol where a key is randomly chosen from a large pool of keys and a random subset of the pool is distributed to the sensor nodes. If two nodes need to communicate, they search their pools to determine whether they are sharing a key or not. If they share a key, it is used to establish a session key. The fully connected network can be obtained, if there is a sufficiently high probability of sharing keys to communicate with other nodes on the network. The disadvantage of the random-key redistribution protocol is that when many nodes are compromised by an attacker, there is a chance of reconstructing the key pool and breaking the scheme.

Time synchronization is very important for sensor network applications. The operations that should be synchronized include sensor scheduling (wake and sleep), sensing tasks, the time-division multiple access control protocol, data aggregation, and multicast authentication. The network time protocol is used on the internet for synchronization purposes. It cannot be used by the sensor network because of hardware constraints. Several other time synchronization protocols for sensor networks have been proposed. The time synchronization algorithm used is similar to reference-broadcast synchronization, a practical timing-sync protocol for sensor networks. It is used to achieve global clock synchronization and pair-wise clock synchronization. Global clock synchronization provides the entire network with clock synchronization. Pair-wise clock synchronization is used for high precision time synchronization between sensor nodes. For synchronization, a receiver node broadcasts a reference packet to help identify clock differences. It is important to note that these synchronization protocols were not designed for hostile environments.

The primary functionality of sensor network nodes is to sense the environment and send sensed data to the base station for further processing. Routing is an important operation in sensor networks. Studies have shown that incorporating security at the design stage is the best way to provide security for the sensor networks. For example, described several security attacks on wireless sensor networks. They analyzed the possible attacks on existing protocols include LEACH and directed diffusion. An efficient and secure routing protocol is proposed in [12] for heterogeneous sensor networks. This protocol can defend against typical attacks on sensor networks. To detect compromised nodes injected by the intruder an efficient algorithm is proposed in [13].

3. ALGORITHM

This section gives an overview of the data collection algorithm that the proposed security algorithm can be used with. The network model of the protocol is depicted in Figure 1. For this approach, the area of the network is divided into inner and outer zones. The energy of the nodes deployed is assumed to be equal. Depending on the number of clusters, the outer zone may be further divided into sectors. After installing the nodes in the area, the next step is configuring the network, through the formation of clusters. The nodes send their location to the base station. The base station is responsible for forming the clusters in two rounds. The base station selects two cluster heads and broadcasts their identities to the network. Each node checks the broadcasted IDs, to see if it has been selected to serve as a cluster head or not. If the node’s ID matches one of the broadcasted ones, then the node is a cluster head for that particular round. The cluster head then broadcasts to other member nodes to join the cluster. Each member node sends an association request, if it seeks to join that particular cluster. While forming the clusters, the balancing of the network is considered to be as an important consideration. From the third round onwards, the previous cluster head chooses the cluster head for the next round. The outer zone algorithm is same as the inner zone for the formation of the clusters.

The data transmission, however, needs to be different in the inner and outer zones. The outer zone, which is farther away from the base station, needs a mobile node to conserve the energy of the fixed sensor nodes. Thus, the outer zone is further divided into sectors and each sector is assigned a mobile node. In the inner zone, the data is sent from the nodes to the base station via the cluster head. In the outer zone, data transmission is relayed via a mobile node. So the
transmission path is from the member nodes to the cluster head, from the cluster head to the mobile node, and from the mobile node to the base station.

Figure 1: Network model of the protocol\textsuperscript{14}.

4. SECURITY CHALLENGES IN WIRELESS SENSOR NETWORKS

This section describes the security challenges faced by a wireless sensor network.

4.1. Data Confidentiality

The system collected data needs to be secured from attackers. In order to achieve confidentiality, the data is encrypted with a secret key when transferred between communicating nodes. However, encrypting the data does not completely solve the problem. If an eavesdropper performs a traffic analysis, it may reveal the key from the cipher text. This makes it possible for the eavesdropper to steal sensitive data. For better confidentiality, several protocols should be followed\textsuperscript{6,15}

1. Sensor readings should not be leaked to other nodes, as the sensor node can have sensitive information stored. To avoid the leakage of sensitive data, nodes should not share keys for decryption and encryption with neighboring nodes\textsuperscript{16}.

2. Secure communication channels need to be created for the WSN.

3. Information about the sensor nodes needs to be encrypted, to the extent possible, for protecting against attacks.

For a mobile WSN, higher risks are involved than for a static WSN, because of the mobility of the mobile node within the network. Therefore, it is important that the mobile node does not share sensor readings, encryption and decryption keys with nodes.

4.2. Data Integrity

Data integrity ensures that the receiver has received the message without any alteration. To achieve data integrity, cryptographic signing and verification techniques are used. Time stamps are used to prevent repaly style attacks.
4.3. Data Authentication

The authentication of messages is important for many applications. It is a process which verifies that the message is from a trusted source\textsuperscript{17}. An adversary can potentially inject packets into the network, so the receiver needs to ensure that the data used in a decision making process originates from the claimed sender. If a sensor node is compromised and it contains the secret key used to broadcast messages, then the adversary can attack the data integrity of the WSN. It is possible to use intrusion detection techniques\textsuperscript{18} to detect such compromised nodes and revoke the broadcast capabilities of the node\textsuperscript{19}.

4.4. Key Establishment

Key establishment is necessary for secure communications between two parties. There are two types of algorithms for key establishment:

1. Symmetric key algorithm
   With symmetric key encryption, the sender and receiver share the same secret key. The sender encrypts the message using the shared secret key and sends it to the receiver. The receiver decrypts the message using the shared secret key.

2. Asymmetric key algorithm
   With an asymmetric algorithm, there are two keys: a private key and public key. The sender encrypts the message using public key and sends it to the receiver. The receiver decrypts the message using the private key of the receiver. Determining the private key from the public key is difficult.

Several key management schemes have been proposed for secure communications between WSN nodes. However, even with this, there is a possibility that if a sensor node is compromised, the network may also become compromised. For instance, in simulations with probabilistic pre-distribution schemes\textsuperscript{20} the compromise of few nodes has been shown to lead to the compromise of the entire network.

4.5. Availability

Availability is a metric related to ensuring that the sensor network is functional throughout the desired network lifetime. Denial of service attacks are one of a number of types of attacks that can result in a failure of availability. There is a serious impact if nodes are not available at the needed point of time. Loss of availability facilitates enemy attacks and invasions, for example, in battlefield surveillance. It impairs other critical applications.

5. ATTACKS ON WIRELESS SENSOR NETWORK

This section describes some of the attacks and threats related to wireless sensor networks.

5.1. Node Capture Attacks

In a node capture attack, an adversary gains physical access over a physical node and can then easily extract its sensitive data. Getting access to a node leads to substantial damage to the entire network. It is presumed that (for many networks) node capture is easy, as there are no physical restrictions to prevent access to the sensor nodes in the network environment\textsuperscript{21}. Researchers proposed different schemes\textsuperscript{7,22-28} for the security of nodes which provide resilience against node capture attacks.

5.2. Denial of Service Attacks

Denial of service attacks diminish the network capacity and expected functionality of the WSN. They can be caused by software errors, hardware failure, malicious broadcasting of high frequency signals, resource exhaustion, and environmental conditions\textsuperscript{21}. If an attack succeeds, the communications system or other functionality may be disabled or limited.
5.3. Software Attacks
In a software attack, an adversary attempts to manipulate the code inside the sensor node by exploiting known vulnerabilities. An example of this type of attack is exploiting a buffer overflow vulnerability. This creates a scenario where data is stored beyond the defined memory limits. If the system is unprotected, this can lead to overwriting code in adjacent memory locations.

5.4. Routing Attacks
Data forwarding and routing are important tasks in wireless sensor networks. Routing protocols have to be robust against node failures and attacks. Many efficient data transfer algorithms have been proposed for use with WSNs. However, most of these algorithms suffer from security issues of various types. Some examples of routing attacks include a black hole attack, spoofed, altered or replayed data attacks, wormhole attacks, selective forwarding attacks, sinkhole attacks, acknowledgment spoofing attacks, and HELLO flood attacks.

5.5. Traffic Analysis Attacks
In a traffic analysis attack, an adversary tries to gather information on the topology of the sensor network and the location of the base station by observing traffic patterns and volumes. There are two types of traffic analysis attacks:

1. Rate monitoring: In this type of attack an adversary monitors the packet sending rates of the nodes in the network and chooses to move towards the node having highest rate.
2. Time correlation monitoring: In this type of attack an adversary eavesdrops on data transmission to identify correlation between sending times to identify communication partners.

6. SECURITY ALGORITHM
An energy efficient algorithm for data collection was proposed in [14]. The operation of a security algorithm for this data collection algorithm is described in detail in the following subsections.

6.1. Assumptions
Several assumptions underlie this discussion. It is assumed that the sensor nodes are installed in untrusted locations. It is assumed that the base station is a trusted source and that it is the gateway to connect to outside world. Because of this, the compromise of the base station would make the entire network useless. Thus, the proposed protocol does not place any trust in the communication infrastructure. It is assumed that the messages are delivered to their receiver with non-zero probability. All the sensor nodes need to trust the base station at creation time. Each node trusts itself for data communication.

6.2. Overview of algorithm
The network is divided into inner and outer zones. There is thus a need for security algorithms for both of these regions. Each is described in detail below. The approach is a form of hybrid security, as the algorithm uses both symmetric and asymmetric cryptography techniques. The inner zone uses symmetric key cryptography techniques due to the minimal energy availability within the sensor nodes. However, the outer zone uses asymmetric key cryptography because the mobile node has significantly more energy available. The complex processing of asymmetric cryptography is performed by the mobile node. The node-to-MDC encryption is similar to the inner zone node-to-cluster head encryption.

6.3. Inner Zone Security
The inner zone security algorithm is comprised of six key principles.

1. Before the installation of the node in the network, the node ID is stored into the node’s memory. Every node in the network, thus, has a unique ID for the transmission of data.
2. The nodes are grouped into clusters. Each cluster has a cluster head for data transmission between cluster group and the base station. The data transmission from the cluster head to the base station is one-hop communication.
3. The base station shares the master secret key with the cluster heads. The master key is selected randomly from a large pool of keys.

4. The node has to transmit data to the cluster head. The message is encrypted using message authentication code (MAC) techniques using a block cipher in counter mode. In the next round, the counter value is incremented.

5. To improve security, the cluster head also uses a pseudorandom function to encrypt the data. A pseudorandom number generator is an algorithm for generating a sequence of numbers to approximate the properties of sequences of random numbers.

6. The receiver has an authenticated value of the key chain it can easily authenticate using the pseudorandom generator.

6.4. Outer Zone Security

The outer zone security algorithm is comprised of seven key principles.

1. In the outer zone the data transmission goes from the cluster head to the base station via an intermediate mobile node.
2. The mobile node is given a unique ID before installation in the network.
3. The mobile node and the base station have significantly more resource and share public keys. The base station shares the master secret key with the cluster heads. The master key is selected randomly from a large pool of keys.
4. The node has to transmit the data to cluster head. The message is encrypted using the message authentication code (MAC) technique using a block cipher in counter mode. In the next round, the counter value gets incremented and generates new encrypted data for better security.
5. The cluster head also uses a pseudorandom function to encrypt the data.
6. The mobile node uses the base station’s public key to encrypt messages.
7. The receiver decrypts the message using its public key, followed by the secret key using the pseudorandom generator.

7. CONCLUSION

Security is critical for many wireless sensor network applications. The limited sensor node memory and energy makes security and privacy a challenging issue. In this paper, a security subsystem was presented for an extremely limited wireless sensor network platform. Most of the security design is universal and applicable to other network of low-end devices. This protocol uses symmetric and asymmetric cryptography techniques. It is designed to become an integral part of practical sensor network implementations. Many security issues for wireless sensor networks remain as open issues and more research is expected in this area.

REFERENCES