Security in Wireless Mesh Networks

Editors

December 15, 2006

Contents

Chapter 1

Key Management in Wireless Sensor Networks

Falko Dressler
Autonomic Networking Group, Dept. of Computer Sciences
University of Erlangen-Nuremberg, Germany
dressler@informatik.uni-erlangen.de

Wireless sensor networks (WSN) and corresponding applications greatly benefit from the proliferation of energy-aware embedded systems. Various application scenarios have successfully shown that the usage of sensor network technology is applicable in different domains. At the same time, the need for security solutions is rising. This includes mechanisms for secure management and control, e.g. routing and software management, as well as for data communication. Similarly, the demand for higher availability including the protection against attacks and misbehaving nodes emerged. Security architectures have been proposed to address these requirements. All these solutions are based on cryptographic algorithms and appropriate key management and key distribution solutions. The objective of this chapter is to provide an overview to state-of-the-art key management and key distribution techniques. Additionally, a classification of key management and key distribution solutions is provided followed by an in-deep study of selected key distribution approaches. The chapter also includes an outlook to application scenarios and outlines the open issues for further research on key management and key exchange.

1.1 Introduction

Wireless sensor networks (WSN) have become a major research domain in the communications community [1]. Besides other issues that have been studied so far [2], energy consumption and security were identified to be the most challenging problem spaces. These properties are influenced by the massively distributed operating principle based on self-organization mechanisms [3]. Similarly, the lifetime of sensor networks [4] depends strongly on the operation mode, i.e. the used routing algorithms, the application behavior, and, finally, the employed security methods.

A survey of security issues in ad hoc and sensor networks can be found in [5]. Additional related work in the security area, focused on WSN, is summarized in [6].

The primary requirements on a successful security architecture are availability, authentication, data confidentiality, integrity, and non-repudiation. Most of these objectives can be addressed using cryptographic hash functions and appropriate encryption schemes. In ad hoc and sensor networks, many proposals were published concerning the use of security measures for particular applications [5]. Security protocols such as [6] define complex architectures to be used in a sensor network environment.

Most of such proposals defer the problem of key management - one of the most sophisticated problems - to be solved elsewhere. Fortunately, several approaches seem to be adequate in this domain as already studied in ad hoc networks [7, 8]. In this chapter, we discuss various key management solutions for sensor networks and provide an overview to general key pre-distribution and proactive key exchange solutions. This survey also provides an classification of key management solutions for wireless sensor networks and an outline of open research issues including efficient public-key encryption in sensor networks [9]. Further discussion on key management solutions can be found in [10].

Besides security architectures and special solutions for routing or key management, the aggregation of encrypted data in WSN was discussed [11] as well as the integration of particular security layers for reliable and secured communication [12]. Finally, secure overlays were proposed to address the security concerns in WSN [13].

In summary, it can be said that many promising proposals can be found in the literature that address the security objectives in sensor networks. Nevertheless, most of these papers only outline the principles or use simulation environments for verification. Experimentation on real sensor nodes is necessary to analyze the behavior of proposed security architectures and to contribute to the sensor network security domain.

All approaches for enabling security in WSN are very scenario dependent. There are different requirements, for example, in an agriculture scenario [14] compared to a habitat monitoring scenario [15]. Other requirements appear in the operation and control domain. Sensor nodes must be reconfigured, calibrated, and reprogrammed [16]. Such operations are very sensible for possible attacks. Finally, it must be mentioned that we ignore the problem of key management. Several solutions have been proposed that address this issue, e.g. [17].

The rest of this chapter is organized as follows. Section 1.2 outlines the major security objectives in sensor networks. Then, section 1.3 discusses application scenarios that strongly depend on security mechanisms and, therefore, profit from efficient and secure key management. This is followed by an overview to key management solutions and mechanisms in section 1.4. Selected key management schemes are presented in detail in section 1.5. Research challenges and open issues in key management are outlines in section 1.6. Finally, section 1.7 concludes the chapter.

1.2 Sensor Network Security Objectives

In this section, we summarize the security properties required by communication networks focusing on the specific capabilities of sensor networks. The necessary security services in sensor networks are not altogether different from those of other networks [5]. The goal of these services is to protect information and resources from attacks and misbehavior. In the context of sensor network security, the following requirements must be ensured for an effective security architecture.

Data confidentiality – Ensures that the transmitted data cannot be understood by anyone other that the desired recipient. Concentrating on sensor networks, it is commonly agreed that the level of necessary confidentiality grows with the concentration or aggregation of multiple sensor measures. Confidentiality is typically enabled by applying either symmetric or asymmetric data encryption techniques. Therefore, keys must be exchanged before a transmission can occur.

Message authentication – Data or message authentication is of paramount importance for many applications in sensor networks. Technically, message authentication ensures the genuineness of received messages. Also covered is data integrity (see below). Usually, cryptographic hash functions using appropriate key material are used to fulfill this objective. In summary, data authentication ensures that received messages were sent by the expected source and not modified during the transmission.

Data integrity – Ensures that the received data was no modified during the transmission. In difference to message authentication, there is no key material involved in processes to ensure data integrity. Similar cryptographic hash functions can be applied in this context. Looking at the properties of sensor networks, data integrity alone is not sufficient due to the inherent property of multi-hop sensor networks that any node can intercept messages, modify them (including the computation of a new hash value), and transmit the modified messages to the final destination.

A detailed analysis of security solutions for WSN is out of scope of this discussion. More information on this topic can be found in [5, 6, 18]. In summary, it can be said that cryptographic hash functions and encryption schemes can be employed to ensure the most prominent security objectives in sensor networks. A prerequisite for this is the exchange of key material. This step must occur before any sensor data can be exchanged.

1.3 Application Scenarios

The security objectives as outlined in the previous section must be considered in various application scenarios for wireless sensor networks. In this section, we summarize selected applications that need to be secured by means of network security solutions. Additionally, we discuss the need for inherently integrating key management solutions into the security approaches in order to validate the efficiency and performance.

One of the first application of network security mechanisms was secure routing in ad hoc and sensor networks [18, 19]. In most routing protocols, routers exchange information on the topology of the network in order to establish routes between nodes. Such information could become a target for malicious adversaries who intend to bring the network down. There are two sources of threats to routing protocols. The first comes from external attackers [20]. By injecting erroneous routing information, replaying old routing information, or distorting routing information, an attacker could successfully partition a network or introduce excessive traffic load into the network by causing retransmission and inefficient routing. The second and also the more severe kind of threats comes from compromised nodes, which advertise incorrect routing information to other nodes. Detection of such incorrect information is difficult: merely requiring routing information to be signed by each node would not work, because compromised nodes are able to generate valid signatures using their private keys. Several solutions have been proposed [18, 21] that all rely on an efficient key management including the detection of compromised or malicious nodes and appropriate revocation mechanisms is strongly demanded.

Similarly, the data dissemination and data forwarding needs to be secured. Proposals such as SPINS [6] address this issue. Key management techniques become even more critical if data must be aggregated, modified, or pre-processed within the network [22, 23]. This case was for example discussed by Castelluccia and co-workers in their study on efficient aggregation of encrypted data in wireless sensor networks [11]. In this case, every node that receives a packet needs to share a key with the sender in order to process the message. Key management can be easily become unserviceable if too many keys need to be stored in each device or if too many nodes become involved in a single hop message exchange.

We discuss this issue later in section 1.5. Higher layer solutions also rely on efficient key management that is assumed to support end-to-end communication as well in a reliable and secure fashion [12].

If software modules are distributed in a sensor network, it must be verified that no attacker might by able to compromise a single node and distribute modified, i.e. infected software modules. Software management solutions for sensor nodes were discussed in several proposals [24, 25, 16]. Key management solutions must provide the basis for secured incremental network programming for wireless sensors [25].

Service discovery is a more generalized form of knowledge distribution. If specific services should be announced and used in a dynamic way, it must be ensured that the identity of the service provider is unambiguous and it has not been compromised so far [26]. A case study for secure distributed service directory for wireless sensor networks outlined the needs on key management solutions [27]. In this context, a secure overlay for service centric sensor networks was proposed [13].

Looking at middleware applications such as service discovery, coordination issues must be considered. Some of the most interesting solutions in the context of ad hoc and sensor networks address security issues including key management objectives as well as particular challenges that emerge in such massively distributed systems. For example, a distributed coordination framework for wireless sensor and actor networks was proposed [28] as well as a cooperation technique for self-organizing mobile ad hoc networks [29].

1.4 Key Management in Sensor Networks

1.4.1 Overview to key management

The organization of key management techniques strongly depends on the selected cryptographic scheme. As mentioned above, we only consider cryptographic hash and encryption mechanisms. In this section, we focus on symmetric schemes that rely on appropriate key exchange and key distribution instead of key verification. In section 1.6, open research challenges, we give an outlook to issues for key management and verification for asymmetric operations.

Key management includes several functionalities. The most prominent, and in several solutions the only one, is key distribution. Nevertheless, key management is also responsible for issues such as key revocation and re-keying. Additionally, it must ensure resiliency to sensor-node capture. All these issues are outlined in section 1.4.2. In this subsection, we present a general classification of key distribution and key exchange solutions.

In theory, key management can be addressed in three ways:

1. Key pre-distribution
2. Pro-active key distribution
3. On-demand key exchange

To date, the only practical option for the distribution of keys to sensor nodes in a large-scale sensor network would have to rely on key pre-distribution [30]. Keys would have to be installed in sensor nodes to accommodate secure connectivity between nodes. However, traditional key pre-distribution offers two inadequate solutions: either a single mission key or a set of separate $n-1$ keys, each being pair-wise privately shared with another node, must be installed in every sensor node. These and more recent solutions that rely on probabilistic schemes [31] or on deployment information [32] are discussed in section 1.4.3.

Pro-active key distribution stands for key exchange after the deployment of the sensor network but before any data communication occurs. Pro-active solutions usually rely on central base stations that provide the necessary key material. On the other hand and to provide more reliability, probabilistic solutions have been proposed that reduce the necessary keys to a minimum but still cover secure communication paths between all nodes [33]. Some of the pro-active key distribution mechanisms also require some pre-deployment actions such as the computation and selection of key rings to be stored in all nodes [30]. Finally, tree-based key distribution algorithms belong to this domain such as [34, 10]. More detailed information on pro-active solutions are provided in section 1.4.4.

Finally, on-demand key exchange mechanisms address the needs of typical applications not to focus on previously exchanged key material but to setup security relations on demand [35]. Public key solutions can be seen to be on-demand solutions as the verification step takes place after the communication was initiated [36]. In general, there are only few approaches available that make use of public-key cryptography. The primary reason are the strong resource limitations in sensor networks, e.g. the computational power or the available memory. Novel approaches that counteract these limitations are still work in progress such [9].

1.4.2 Key management issues

In this subsection, we present the basic features of key management solutions. All solutions for key management, basically concentrate on key distribution or key pre-distribution. Nevertheless, issues such as revocation and re-keying must be considered as well.

Key distribution – Key distribution is the basis of all key management schemes [30]. It can be solved either by key pre-distribution prior to deployment or pro-active in a sensor network prior to any data communication. Key distribution is the main topic of this chapter and is outlined in the following subsections.

Revocation – When a sensor node is compromised, it is essential to be able to revoke keys associated with this sensor node. This may involve a complete new key distribution in case of a single mission key. Usually, only the according key rings need to be discarded and re-build. Revocation procedures rely on an agreement that defines which keys need to be discarded. In most schemes, a controller node coordinates such a process. If there is no central controller available, election algorithms are used to select a node that performs the necessary tasks.

Re-keying – The lifetime of (particular) keys can be limited using expiration times. Although, such mechanisms are rarely used in sensor networks, the expiration of keys and the necessary re-keying is a fundamental function in key management solutions. Basically, re-keying is equivalent with a self-revocation of a key by a node. It involves all nodes that share the specific key. Re-keying schemes were categorized into two classes: stateful and stateless [17].

Resiliency to sensor-node capture – The unattended operation of sensor nodes in hostile areas raises the possibility of sensor-node capture. Although, node capture is a general threat that affects all security mechanisms, key management solutions must be aware of such situations and provide adequate mechanisms to counteract such captures. Basically, similar mechanisms as for general key revocation can be used in this case.

1.4.3 Key pre-distribution

Traditional Internet-based key exchange and key distribution protocols require an infrastructure providing trusted third parties. Such approaches are no feasible for large-scale sensor networks due to the following reasons: the network topology is not known prior to deployment, the communication range is very limited, and the networks are dynamic in terms of sleep cycles or even node failures. Therefore, most key management approaches are based on *key pre-distribution*. Keys would have to be installed in sensor nodes to accommodate secure connectivity between nodes. Figure 1.1 depicts well-known key pre-distribution schemes. The intention of key pre-distribution is to make key material available during or before the deployment in order to minimize subsequent cryptographic overhead for key generation. In the following, the mentioned schemes are explained and discussed.

Single mission key – This approach deals with a pre-installed key on all sensor nodes. Usually, this key cannot be changed and lasts for the whole lifetime of the network. Depending on the scenario, a single mission key might be a feasible approach considering a small network that needs to perform an application with a limited runtime. In any other case, such a solution is inadequate because the capture of any single node may compromise the complete network. Additionally, attacks can be initiated to recover the key using eavesdropped packets. Because all nodes use the same key, an attacker will be able to collect enough data for such an attack in quite a short time. The selective revocation is not possible in this scenario.

Set of $n-1$ keys – In contrast to the single mission key approach, the pair-wise private sharing of keys between every two sensor nodes avoids the compromising of the entire sensor network upon node capture since selective key revocation becomes possible. However, this solution requires pre-distribution and storage of $n-1$ keys in each sensor node and $n(n-1)/2$ per sensor network. It was shown in [30] that this approach is impractical for sensor networks consisting of more than 10,000 nodes, for both intrinsic and technological reasons. First, pair-wise private key sharing between any two sensor nodes would be unusable since direct node-to-node communication is achievable only in small node neighborhoods delimited by communication range and sensor density. Secondly, incremental addition and deletion as well as re-keying of sensor nodes would become both expensive and complex as they would require multiple keying messages to be broadcast network-wide to all nodes during their non-sleep periods (i.e., one broadcast message for every added/deleted node or re-key operation). Third, a dedicated RAM memory for storing $n-1$ keys would push the on-chip, sensor-memory limits for the foreseeable future, even if only short, 64 bit, keys are used and would complicate fast key erasure upon detection of physical sensor tampering. More scalable approaches in this context were proposed in [30, 37].

Random pre-distribution – The overhead due to the storage requirements for $n(n-1)/2$ keys can for example be reduced using randomized techniques. Instead of storing the whole key ring for all $n \times n$ communication relationships, only samples of the complete key ring are stored in each sensor node. To simplify the deployment of the sensor network as well as to allow the adding of nodes at any time without the necessity of key exchange procedures, probabilistic methods can be used to choose part of the key ring for each sensor. Such scenarios were investigated by several groups [31, 30, 38]. The complexity of such approaches does not lie in the key management but in the identification of paths through the network that represent trusted chains. In such a chain, two neighboring nodes must share identical keys out of their key ring samples. So the problem of key distribution can be reduced to the problem of path finding or routing. Specific solutions using random subset assignment and grid assignment techniques were studied in [39].

Pre-distribution using deployment knowledge – Finally, another approach can be used to reduce the storage requirements known from the set of $n-1$ key solutions, the use of state information. Such solutions exploit the deployment knowledge, i.e. the state of the sensors, to avoid unnecessary key assignments and to reduce the number of required keys that each sensor node should carry. At the same time, it is possible to support higher connectivity and better resilience against node failures. In this context, state information means the classification of sensor node states into active and sleep [32, 40]. Using this information, the efficiency of pure probabilistic schemes can be noticeable improved.

1.4.4 Pro-active key distribution

In contrast to key pre-deployment strategies, *pro-active key distribution* schemes are based on dynamic key generation or key exchange algorithms, respectively. Most of these approaches need to be initialized by a key pre-deployment mechanism as described above. Afterward, keys can be generated and replaced dynamically. It must be mentioned that the dynamics in pro-active solutions is limited. Compared to on-demand algorithms that can create new key just in time with an forthcoming communication [35], pro-active mechanisms need to be executed prior any data communication, i.e. before the key material might be needed. Figure 1.2 depicts an overview of typical pro-active key distribution methodologies. In the following, possible solutions for such schemes are discussed.

Base station approach – Bootstrapping any further secured communication can be initiated by selected base stations. Considering typical sensor network architectures, base stations are used to provide connectivity between the sensor network and a fixed communication infrastructure. Therefore, compromising the base station could render the entire sensor network useless. Thus, the base stations are a necessary part of the trusted computing base [6]. A trust setup mimics this and so all sensor nodes intimately trust the base station: at creation time, each node is given a master key, which is shared with the base station. All other keys are derived from this key.

Probabilistic key sharing – Another solution space is again based on probabilistic schemes.

Initially, trust is created by the use of subsets of key rings. The subsets can be either *balanced*, i.e. each node is required to store the same amount of keys [30]. This procedure results in a homogeneous distribution of both, keys and subsequent processing requirements due to key management actions. Depending on the topology of the sensor network and the communication relationships, e.g. arbitrary communication vs. base station solutions, this approach can lead to unfair exhaustion of resources of single sensor nodes. Additionally, heterogeneity of sensor nodes cannot be exploited, e.g. if the network consists of small nodes with very limited resources and larger ones that are able to store huge amounts of keys. *Unbalanced* approaches have been discussed that promise to solve this problem [33].

Tree-based key management – In many sensor network scenarios, either the communication can be compared to a tree with a single base station or gateway at the root [9] or the deployment follows a hierarchical structure [10]. In both cases, the key management can be adapted to the tree structure in order to reduce the number of keys that need to be pre-distributed or pro-actively computed.

1.5 Selected Key Management Schemes

In this section, we provide more details on selected key management schemes. Again, we follow the classification presented in the previous section. Many proposed solutions are constructed on top of each other. Therefore, we try to follow the chronological order as well. The first three methods, i.e. balanced random pre-distribution, unbalanced random pre-distribution, and state-based pre-distribution, can directly be compared in terms of $p(\lambda)$, the probability that two sensors share at least one key after the pre-distribution phase. This parameter is outlined in each subsection. Afterward, tree-based key distribution is discussed.

1.5.1 Balanced random pre-distribution in homogeneous networks

Eschenauer and Gligor presented a scheme for key management in distributed sensor networks using probabilistic key sharing and a simple protocol for shared-key discovery and path-key establishment, and for key revocation, re-keying, and incremental addition of nodes [30]. Here, we discuss the three phases key pre-distribution, shared-key discovery, and path-key establishment.

The *key pre-distribution* phase consists of five off-line steps:

- generation of a large pool of P keys (e.g., 2^{17} - 2^{20} keys) and of their key identifiers
- random drawing of k keys out of P without replacement to establish a key ring of a sensor
- loading the key ring into the memory of each sensor node
- saving key identifiers of a key ring and associated sensor identifier on a trusted controller node
- for each node, loading the i-th controller node with the key shared with that node

This procedure ensures that only a small number of keys need to be placed on each sensor node's key ring to ensure that any two sensor nodes share at least a key with a chosen probability

The *shared-key discovery* phase takes place during the sensor network initialization. where every node discovers its neighbors in the wireless communication range with which it shares keys. The simplest way to discover neighboring nodes that share a key with a specific node is to broadcast, in clear text, the list of identifiers of the keys on the local key ring. Therefore, this phase establishes the topology of the sensor network as seen by the network layer. A link between any two neighboring nodes exists if they share a key. The other way around, if a link exists between two nodes, all communication between these nodes can be secured using appropriate cryptographic algorithms.

The *path-key establishment* phase finally assigns a path-key to selected pairs of nodes that do not share a key but are connected by two or more links at the end of the shared-key discovery phase.

Using random graph theory, Eschenauer and Gligor have shown that, given a pool of

P keys and randomly choosing k keys for the key ring, the probability p of sharing a key between any two nodes in a neighborhood can be calculated as follows:

$$\begin{aligned} p &= 1 - Pr[\text{two nodes do not share any key}] \\ &= 1 - \frac{((P-k)!)^2}{(P-2k)!P!} \end{aligned} \tag{1.1}$$

In [30], the following numerical example was depicted. At us assume a sensor network consisting of $n = 10,000$ nodes and a desired probability of $P_c = 0.99999$ for obtaining an "almost certainly" connected network, and a wireless communication range that allows the neighborhood connectivity of 40 nodes. Then $k = 250$ out of $P = 100,000$ keys must be stored in each node. If the connectivity increases to 60, only 200 keys are needed.

1.5.2 Unbalanced random pre-distribution in heterogeneous networks

Traynor and co-workers demonstrated that a probabilistic unbalanced distribution of keys throughout the network that leverages the existence of a small percentage of more capable sensor nodes can not only provide an equal level of security but also reduce the consequences of node compromise. They demonstrated the effectiveness of this approach on small networks using a variety of trust models and then demonstrated the application of this method to very large systems [33].

As shown in the previous subsection, random key pre-deployment in sensor networks has assumed very large random-graph arrangement such that all neighbors within the transmission radius of a given node are reachable. Communication between adjacent nodes is therefore limited only by key matching. This model is not always realistic for a number of reasons. In the unbalanced case, the network now consists of a mix of nodes with different capabilities and missions. The sensing or Level 1 (L1) nodes are assumed to be very limited in terms of memory and processing capability, and perform the task of data collection. Level 2 (L2) nodes have more memory and processing ability. These nodes are equipped with additional keys, and take on the role of routers and gateways between networks.

Again, the connectivity must be analyzed. In the following, n is the number of L1 nodes in a neighborhood, and g is the number of L2 nodes in a neighborhood, where applicable. The scheme for the unbalanced distribution of keys throughout a wireless sensor network builds upon the previously described balanced approach of Eschenauer and Gligor. Given the same generated key pool of size P, we store a key ring of size k keys in each sensor (L1) node, and a key ring of size m keys in each L2 node, where $m \gg k$. Then, the probability of an L2 and L1 having at least one key in common can be calculated as follows:

$$\begin{aligned} p &= 1 - Pr[\text{two nodes do not share any key}] \\ &= 1 - \frac{(P-k)!(P-m)!}{(P-m-k)!P!} \end{aligned} \tag{1.2}$$

Traynor and co-workers demonstrated that their unbalanced approach has similar security capabilities as the balanced case. In a simulation, they have proven that a key ring of 328 keys (considering 40 neighboring nodes) is comparable to 5 L2-nodes with 711 keys and 35 L1-nodes with 30 keys respectively. Therefore, they achieved a noticeable reduction of the load of typical sensor nodes by exploiting heterogeneous sensor network environments. Additionally, the unbalanced scheme not only reduces the number of transmissions necessary to establish session-keys but also reduces the effects of both single and multiple node captures. Lastly, the unbalanced scheme allows for even the most memory constrained platforms, from sensor nodes to RFID tags, to hold enough keys to establish secure connections for communication.

1.5.3 State-based key pre-distribution supporting busy-sleep cycles

Location information can be facilitated as deployment knowledge for improvement of the previously discussed key pre-distribution schemes. If two sensor nodes are closely located to each other, they have very low probability to be in active-state at the same time. Therefore, unnecessary key assignments can be eliminated since keys shared only between such closely located nodes may be hardly used. In [32, 40], Park and co-workers propose a random key pre-distribution scheme that exploits new deployment knowledge, the state of the sensors, to avoid unnecessary key assignments and to reduce the number of required keys that each sensor node must carry while supporting higher connectivity and better resilience against node captures.

In figure 1.3, an example is shown for key assignments in a sensor network. s_i and k_j (with $i = 1, 2, ...$ and $j = 1, 2, ...$) denote the sensor nodes and their pre-distributed keys, respectively. Let T_i denote the time-interval when sensor s_i is supposed to be in active-state with high probability. Two sensors, s_1 and s_2, are deployed closely, so they may share more keys as proposed in [32]. Suppose that s_1 and s_2 have key set $\{k_1, k_2, k_3, k_4\}$ and $\{k_1, k_3, k_5, k_6\}$, respectively. During T_1, s_1 and s_2 are in active-state and sleep-state, respectively. Then, as time goes by, s_1 and s_2 transit their states to sleep and active, respectively. If s_1 and s_2 are in active state at the same time with very low probability, the shared key only between them, $\{k_1, k_3\}$, may be hardly used. Therefore, the key assignments of these keys to s_1 and s_2 are unnecessary.

Park an coworkers used this idea to develop a state-based key management scheme [40]. They assumed that sensor nodes are implemented to be in active-state at specific time-intervals with high probability and in other time-intervals the probability is relatively low. Then, sensor nodes can be grouped by the time intervals when they have high probabilities to be in active-state. For instance, if sensor s_1 has high probability to be in active-state at time-interval T_1, it may be grouped within the first group. Using these assumptions, the active-state group (ASG) can be defined as the group of sensor nodes with high probability to be in active-state at the same time interval. The calculation of the active-probability is depicted in [40].

For key distribution, Park *et al.* use two key pools:

- Global Key Pool (GlP): A GlP S is a pool of random symmetric keys, from which a group key pool is generated. The cardinality of equals to $|S|$.
- Group Key Pool (GrP): A GrP S_i is a subset of GlP S for i-th group, from which a key ring is generated. The cardinality of S_i equals to $|S_G|$.

These pools are used for the key pre-distribution phase. Assuming L groups defined during the modeling of the ASG, the key server generates a large GlP S and divides it into L GrPs

S_i for each ASG G_i. The purpose of setting up the GrP is to allow the time-neighbor ASGs to share more keys. After completing the GrP setup, for each sensor node j in ASG G_i, a randomly selected key ring $R_{j,i}$ from its corresponding GrP S_i is loaded into the memory of the sensors. For the assignment, an overlapping factor a is used that determines a certain number of common keys between two nearby time-interval groups. Since keys selected from the other groups are all distinct, the sum of all the number of keys should be equal to $|S|$. Therefore, $|S_G|$ can be calculated as follows:

$$|S_G| = \frac{|S|}{L - aL + a} \tag{1.3}$$

The probability that two sensors share at least one common key can be expressed as $1 - Pr$[two nodes do not share any key]. Since the size of GrP is $|S_G|$, the number of keys shared between two GrPs is $\lambda|S_G|$, where is λ is 1, a, or 0. According to the value of λ, we should consider three cases for finding the required probability; two sensors come from same group ($\lambda = 1$), the neighbor two groups ($\lambda = a$), and the different groups which are not neighbor each other ($\lambda = 0$). The same overlapping key pool method used in [32] can be adopted. The first node selects i keys from the $\lambda|S_G|$ shared keys, it then selects the remaining $R - i$ keys from the non-shared keys. The second node selects R keys from the remaining $|S_G| - i$ keys from its GrP. Therefore, $p(\lambda)$, the probability that two sensors share at least one key when their GrPs have $\lambda|S_G|$ keys in common, can be calculated as:

$$\begin{aligned} p(\lambda) &= 1 - Pr[\text{two nodes do not share any key}] \\ &= 1 - \frac{\sum_{i=0}^{min(R,\lambda|S_G|)} \begin{pmatrix} \lambda|S_G| \\ i \end{pmatrix} \begin{pmatrix} (1-\lambda)|S_G| \\ R-i \end{pmatrix} \begin{pmatrix} |S_G| - i \\ R \end{pmatrix}}{\begin{pmatrix} |S_G| \\ R \end{pmatrix}^2} \end{aligned} \tag{1.4}$$

A detailed performance analysis of this approach is presented in [40]. In many scenarios, this scheme offers a better performance compared to the approaches from Eschenauer *et al.* and Du *et al.*

1.5.4 Tree-based key distribution

Chen and Drissi contributed to the pro-active key management by arranging the sensor nodes in a hierarchical form [10]. They express the communication in a sensor network in a well-structured way and provide several application examples that support and confirm this approach. Given such a hierarchical design of a sensor network as depicted in figure 1.4, two forms of communication are necessary: between neighboring nodes at the same level n (and the same group) and between sensors and their direct leaders in the next higher level $n + 1$.

Appropriate keys must be distributed according to the communication paths in the network. Chen *et al.* propose the following scheme in which all nodes (except leaves and the root) are given four types of keys, namely the group key (only one), the uplevel pair-wise key (only one), the downlevel group key (only one), and the downlevel pair-wise key (can be many). These keys and their usage is described in the following. Hereby, we follow the notation as used in figure 1.4.

- Group key – The group key must be known by each group member in order to communication in the direct neighborhood, i.e. in the local group. Examples are nodes A and B, C and D, and F and G, respectively. A and B belong to the same group. Therefore, they must share the key $K_G\{A, B\}$ for secure communication. This group key must also be known by the direct group leader, i.e. node F in our example. This knowledge is used for key management and command issues instead of data communication.
- Downlevel group key – The downlevel group key is the same key as the group key described above. This key is only used for command purposes, e.g. key management issues for sensor node addition, replacement, and deletion.
- Uplevel pair-wise key – Communication between disjunctive groups must occur via the network-inherent hierarchy, e.g. communication between A and C must use node F as a gateway. Therefore, each sensor node must share a private key with its uplevel group leader. Examples are pair-wise keys $K\{A, F\}$ between nodes A and F and $K\{F, H\}$ between F and H.

  item Downlevel pairwise key – This key was is the same as the uplevel pair-wise key but seen from the different angle.

As already mentioned, the communication paths follow the hierarchy as do the key sharings. If node A wants to send a message to D, the following transmissions will occur: A→F using $K\{A, F\}$, F→G using $K_G\{F, G\}$, and G→D using $K\{D, G\}$.

Considering the performance of this approach, we examine the amount of keys that is necessary for communication and key management in such a hierarchical design. As described in [10], a network of n sensor nodes with a depth of the tree of d (assuming a complete tree) results in $\log_d n$ sensor nodes per group. Each leaf sensor only needs to store two keys, the root sensor needs to store approximately $\log_d n + 1$ keys. All the other nodes need to store about $\log_d n + 3$ keys. Therefore, the key storage requirement is $O(\log_d n)$.

A similar tree-based approach for secure key distribution is described by Blaß *et al.* [34]. In this work, the primary objective is on securely integrating new nodes in an existing tree. Additionally, the hierarchical structure is not based on a pre-defined setup but on the real communication paths that can be observed in the network.

1.6 Open Research Challenges

The typical hardware and software constraints make it impractical to use the majority of the current secure algorithms, which were designed for powerful workstations. For example, the working memory of a sensor node is insufficient to even hold the variables (of sufficient length to ensure security) that are required in asymmetric cryptographic algorithms (e.g., RSA and Diffie-Hellman), let alone perform operations with them [6]. A particular challenge is broadcasting authenticated data to the entire sensor network. Current proposals for authenticated broadcast are impractical for sensor networks. First, most proposals rely on asymmetric digital signatures for the authentication, which are impractical for multiple reasons (e.g. long signatures with high communication overhead of 50-1000 bytes per packet, very high overhead to create and verify the signature). The main problem of any public key based security system is to make each users public key available to others in such a way that its authenticity is verifiable. In mobile ad hoc networks, this problem becomes even more difficult to solve because of the absence of centralized services and possible network partitions. More precisely, two users willing to authenticate each other are likely to have access only to a subset of nodes of the network (possibly those in their geographic neighborhood). Self-organized public key management is a first approach to address the security requirements in a scalable way [36]. On the other hand, cryptographic primitives are the fundamental building blocks of every secure protocol the knowledge of algorithm usability is crucial for the design of new protocols for sensor networks. More acceptable encryption schemes using elliptic curve cryptography are proposed in [9].

Broadcast authentication is another problem. Even previously proposed purely symmetric solutions for broadcast authentication are impractical: Gennaro and Rohatgi's initial work required over 1 kByte of authentication information per packet [41], and Rohatgi's improved k-time signature scheme requires over 300 bytes per packet [42]. Perrig *et al.* implemented the necessary primitives [6]. The available computational resources are usually very limited and often not concerned security solutions. A typical performance evaluation must employ adequately calibrated simulation models [43]. In this reference, measurements of typical sensor nodes are depicted that show that even symmetrical cryptography has practical limitations in real sensor networks.

A common characteristic of sensor networks is their severely limited energy supply. Ultimately, the available energy determines that, for example, base stations differ from nodes in having longer-lived energy supplies and having additional communications connections to outside networks. In order to minimize the energy usage, a security subsystem should place minimal requirements on the processor, and add minimal information to each message transmitted. On the other hand, the limited lifespan of each node limits the life time of usable keys. Given the severe hardware and energy constraints, we must be careful in the choice of cryptographic primitives and the security protocols in the sensor networks.

Key agreement is necessary based on scalable and efficient solutions. In [44], three approaches to the problem of user-friendly key agreement (and mutual authentication) in settings where the users do not share any authenticated information in advance were proposed.

The first approach belongs to the family of solutions requiring the users to compare strings of words, whereas the other two approaches are based on radio channel specific techniques, namely, distance-bounding and integrity-codes (I-codes). Scalable key management with inherent self-configuration will allow the deployment of even larger networks [45].

Last but not least, group key management including group re-keying mechanisms for sensor networks are needed. Most existing group re-keying schemes are not suitable for sensor networks since they have large overhead and are not scalable. This problem was addressed by a family of pre-distribution and local collaboration-based group re-keying (PCGR) schemes [17]. These schemes are designed based on the ideas that future group keys can be preloaded to the sensor nodes before deployment, and neighbors can collaborate to protect and appropriately use the preloaded keys.

In summary, the following research aspects and challenges for key management solutions can be formulated:

- energy-aware key management
- public key management (key infrastructure)
- feasible public key cryptography
- key agreement mechanisms
- group key management

1.7 Conclusion

Security issues in wireless sensor networks have been studied by various groups in order to fulfill the raising demands of applications in this domain. In these works, special requirements on security solutions have been identified that are correlated to the specific characteristics of sensor networks (strongly limited resources in terms of processing and storage capacity, communication bandwidth, and energy). Based on the results, many proposals for security in WSN are available that focus on routing, data aggregation, and cooperation issues. All of them rely on appropriate key management solutions that must be made available for sensor network installations.

In this chapter, we presented an overview to key management and key distribution approaches for application in wireless sensor networks. We started with a first categorization of key management solutions in the area of WSN. Basically all proposals are based on efficient key pre-distribution or pro-active key exchange supporting symmetric cryptographic techniques. The different classes can be distinguished by the presumed knowledge about network topology and routing mechanisms.

Based on this classification, we described selected examples in detail in order to demonstrate the basic principles of the available solutions. We added a brief discussion on the performance to each of these mechanism.

Besides a few academic proposals and testbeds, asymmetric solutions cannot be found in sensor networks. There are two reasons for this observation: first, asymmetric cryptographic operations cannot be efficiently used in small embedded systems and, secondly, to date there is no public key infrastructure available for use in wireless sensor networks.

Finally, we also provided a section outlining open issued and challenges in the domain of security in WSN focusing on key management. This roundup is intended to motivate further research work in this domain.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–116, August 2002.

[2] C.-Y. Chong and S. P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, August 2003.

[3] F. Dressler, "Self-Organization in Ad Hoc Networks: Overview and Classification," University of Erlangen, Dept. of Computer Science 7, Technical Report 02/06, March 2006.

[4] F. Dressler and I. Dietrich, “Lifetime Analysis in Heterogeneous Sensor Networks,” in *9th EUROMICRO Conference on Digital System Design - Architectures, Methods and Tools (DSD 2006)*, Dubrovnik, Croatia, August 2006, pp. 606–613.

[5] D. Djenouri and L. Khelladi, “A Survey of Security Issues in Mobile Ad Hoc and Sensor Networks,” *IEEE Communication Surveys and Tutorials*, vol. 7, no. 4, pp. 2–28, December 2005.

[6] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, “SPINS: Security Protocols for Sensor Networks,” *Wireless Networks*, vol. 8, no. 5, pp. 521–534, September 2002.

[7] B. Wu, J. Wu, E. B. Fernandez, and S. Magliveras, “Secure and Efficient Key Management in Mobile Ad Hoc Networks,” in *1st International Workshop on Systems and Network Security (SNS 2005)*, April 2005.

[8] J.-P. Hubaux, L. Buttyan, and S. Capkun, “Security, testbeds and applications: The quest for security in mobile ad hoc networks,” in *ACM International Symposium on Mobile and Ad Hoc Networks (ACM MobiHoc)*, October 2001.

[9] E.-O. Bla and M. Zitterbart, “Towards Acceptable Public-Key Encryption in Sensor Networks,” in *The 2nd International Workshop on Ubiquitous Computing (ACM SIGMIS)*, May 2005.

[10] X. Chen and J. Drissi, “An Efficient Key Management Scheme in Hierarchical Sensor Networks,” in *2nd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2005): International Workshop on Wireless and Sensor Networks Security (WSNS’05)*, Washington, DC, USA, November 2005.

[11] C. Castelluccia, E. Mykletun, and G. Tsudik, “Efficient Aggregation of Encrypted Data in Wireless Sensor Networks,” in *Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous 2005)*, July 2005, pp. 109–117.

[12] F. Dressler, “Reliable and Semi-reliable Communication with Authentication in Mobile Ad Hoc Networks,” in *2nd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2005): International Workshop on Wireless and Sensor Networks Security (WSNS’05)*, Washington, DC, USA, November 2005, pp. 781–786.

[13] H.-J. Hof, E.-O. Bla, and M. Zitterbart, “Secure Overlay for Service Centric Wireless Sensor Networks,” in *First European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS 2004)*, August 2004.

[14] A. Baggio, “Wireless sensor networks in precision agriculture,” in *ACM Workshop on Real-World Wireless Sensor Networks (REALWSN 2005)*, Stockholm, Sweden, June 2005.

[15] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, “Wireless Sensor Networks for Habitat Monitoring,” in *First ACM Workshop on Wireless Sensor Networks and Applications*, Atlanta, GA, USA, September 2002.

[16] G. Fuchs, S. Truchat, and F. Dressler, “Distributed Software Management in Sensor Networks using Profiling Techniques,” in *1st IEEE/ACM International Conference on Communication System Software and Middleware (IEEE/ACM COMSWARE 2006): 1st International Workshop on Software for Sensor Networks (SensorWare 2006)*, New Dehli, India, January 2006, pp. 1–6.

[17] W. Zhang and G. Cao, “Group Rekeying for Filtering False Data in Sensor Networks: A Predistribution and Local Collaboration-Based Approach,” in *24th IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM 2005)*, March 2005, pp. 503–514.

[18] L. Zhou and Z. J. Haas, “Securing Ad Hoc Networks,” *IEEE Network*, vol. 13, no. 6, November/December 1999.

[19] B. R. Smith, S. Murphy, and J. J. Garcia-Luna-Aceves, “Securing distance-vector routing protocols,” in *Symposium on Network and Distributed Systems Security*, Los Alamitos, CA, February 1997, pp. 85–92.

[20] C. Karlof and D. Wagner, “Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures,” in *Workshop on Sensor Network Protocols and Applications*, 2003.

[21] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. Belding-Royer, “A Secure Routing Protocol for Ad Hoc Networks,” in *International Conference on Network Protocols (ICNP)*, November 2002.

[22] L. Hu and D. Evans, “Secure aggregation for wireless sensor networks,” in *Workshop on Security and Assurance in Ad hoc Networks*, 2003.

[23] B. Przydatek, D. Song, and A. Perrig, “SIA: Secure Information Aggregation in Sensor Networks,” in *ACM SenSys*, 2003, pp. 255–265.

[24] D. Culler, J. Hill, P. Buonadonna, R. Szewczyk, and A. Woo, “A Network-Centric Approach to Embedded Software for Tiny Devices,” in *First International Workshop*

on Embedded Software (EMSOFT 2001), Tahoe City, CA, USA, October 2001.

[25] J. Jeong and D. Culler, “Incremental Network Programming for Wireless Sensors,” in *First IEEE International Conference on Sensor and Ad hoc Communications and Networks (IEEE SECON)*, June 2004.

[26] F. Almenarez and C. Campo, “SPDP: A Secure Service Discovery Protocol for Ad-Hoc Networks,” in *9th Open European Summer School and IFIP Workshop on Next Generation Networks*, Budapest, Hungary, 2003.

[27] H.-J. Hof, E.-O. Bla, T. Fuhrmann, and M. Zitterbart, “Design of a Secure Distributed Service Directory for Wireless Sensornetworks,” in *First European Workshop on Wireless Sensor Networks*, January 2004.

[28] T. Melodia, D. Pompili, V. C. Gungor, and I. F. Akyildiz, “A Distributed Coordination Framework for Wireless Sensor and Actor Networks,” in *6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM Mobihoc 2005)*, Urbana-Champaign, Il, USA, May 2005, pp. 99–110.

[29] L. Buttyn and J.-P. Hubaux, “Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks,” *Mobile Networks and Applications*, vol. 8, no. 5, pp. 579–592, October 2003.

[30] L. Eschenauer and V. D. Gligor, “A Key-Management Scheme for Distributed Sensor Networks,” in *9th ACM Conference on Computer and Communication Security (ACM CCS)*, Washington, DC, November 2002.

[31] H. Chan, A. Perrig, and D. Song, “Random Key Management Predistribution Schemes for Sensor Networks,” in *IEEE Symposium on Research in Security and Privacy*, 2003.

[32] W. Du, J. Deng, Y. S. Han, S. Chen, and P. Varshney, “A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge,” in *IEEE Infocom 2004*, March 2004, pp. 586–597.

[33] P. Traynor, H. Choi, G. Cao, S. Zhu, and T. L. Porta, “Establishing Pair-Wise Keys in Heterogeneous Sensor Networks,” in *25th IEEE Conference on Computer Communications (IEEE INFOCOM 2006)*, Barcelona, Spain, April 2006.

[34] E.-O. Bla, M. Conrad, and M. Zitterbart, “A Tree-Based Approach for Secure Key Distribution in Wireless Sensor Networks,” in *The REALWSN*, June 2005.

[35] N. Asokan and P. Ginzboorg, “Key Agreement in Ad Hoc Networks,” *Computer Com-*

mmunications, vol. 23, pp. 1627–1637, 2000.

[36] S. Capkun, L. Buttyn, and J.-P. Hubaux, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52–64, January 2003.

[37] W. Du, J. Deng, Y. S. Han, and P. Varshney, "A Pairwise Key Predistribution Scheme for Wireless Sensor Networks," in *10th ACM Conference on Computer and Communications Security (CCS)*, October 2003, pp. 42–51.

[38] S. Zhu, S. Xu, S. Setia, and S. Jajodia, "Establishing pair-wise keys for secure communication in ad hoc networks: A probabilistic approach," in *IEEE International Conference on Network Protocols (ICNP)*, November 2003.

[39] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *10th ACM Conference on Computer and Communications Security*, Washington D.C., USA, October 2003, pp. 52–61.

[40] J. Park, Z. Kim, and K. Kim, "State-based Key Management Scheme for Wireless Sensor Networks," in *2nd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2005): International Workshop on Wireless and Sensor Networks Security (WSNS'05)*, Washington, DC, USA, November 2005.

[41] R. Gennaro and P. Rohatgi, "How to sign digital streams," in *Advances in Cryptology - Crypto'97*, vol. LNCS 1294, Berlin, Germany, 1997, pp. 180–197.

[42] P. Rohatgi, "A compact and fast hybrid signature scheme for multicast packet authentication," in *6th ACM Conference on Computer and Communication Security*, November 1999.

[43] M. Passing and F. Dressler, "Experimental Performance Evaluation of Cryptographic Algorithms on Sensor Nodes," in *3rd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2006): 2nd IEEE International Workshop on Wireless and Sensor Networks Security (WSNS'06)*, Vancouver, Canada, October 2006, pp. 882–887.

[44] M. Cagalj, S. Capkun, and J.-P. Hubaux, "Key agreement in peer-to-peer wireless networks," *Proceedings of the IEEE (Special Issue on Cryptography and Security)*, vol. 94, no. 2, pp. 467–478, February 2006.

[45] F. Liu and X. Cheng, "A Self-Configured Key Establishment Scheme for Large-Scale

Sensor Networks," in *3rd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2006)*, Vancouver, Canada, October 2006, pp. 447–456.

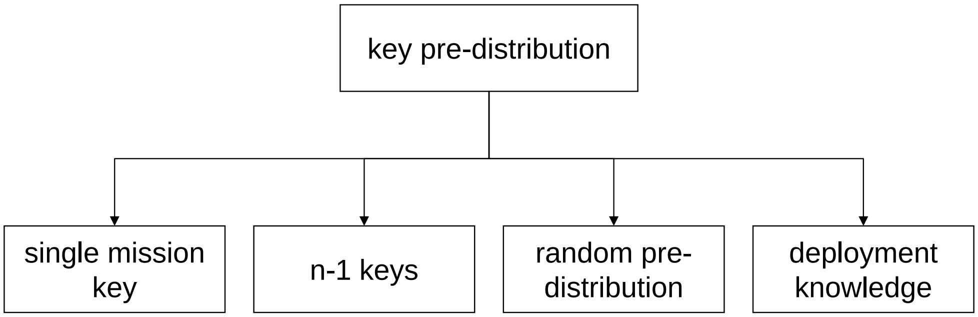

Figure 1.1: Overview to key pre-distribution techniques

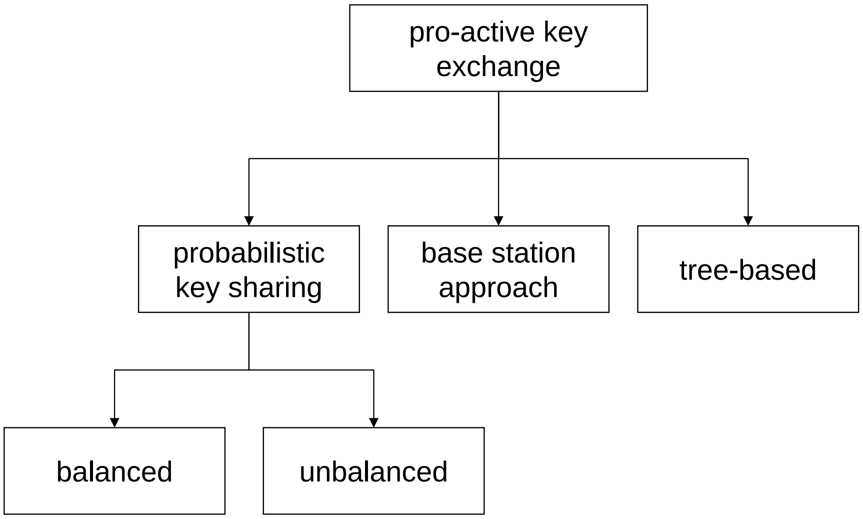

Figure 1.2: Pro-active key management techniques

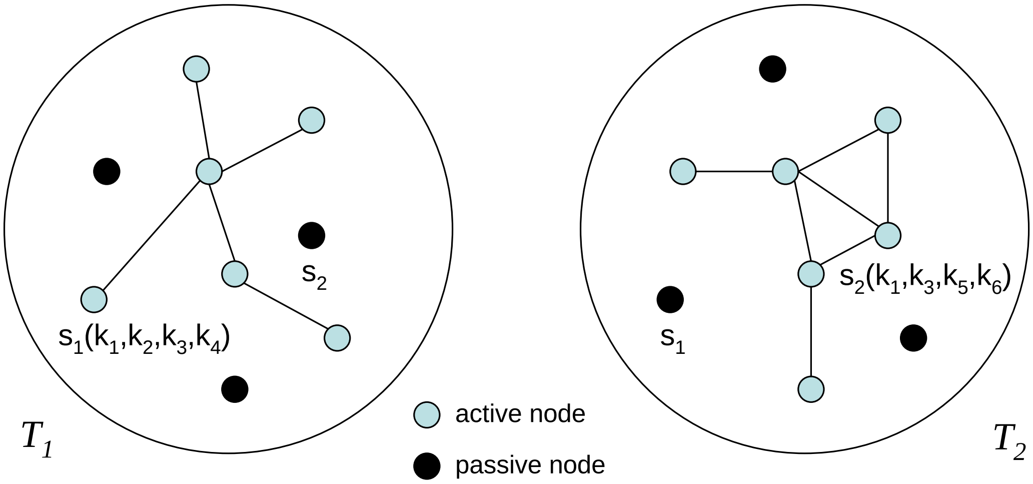

Figure 1.3: Typical key assignments in sensor networks monitored at time T_1 and T_2

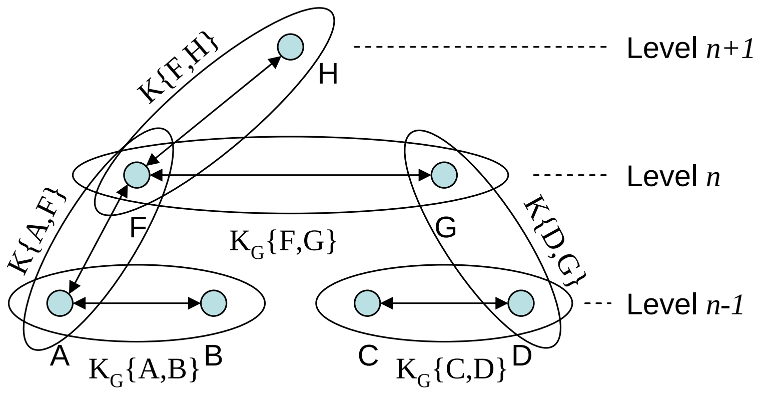

Figure 1.4: Hierarchical or tree-based organization of sensors and the according keys