Key Recoverability in Wireless Sensor Networks

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 10, 2019, accepted October 28, 2019, date of publication November 11, 2019, date of current version November 21, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2952945*

# Key Recoverability in Wireless Sensor Networks

**FILIPPO GANDINO, (Member, IEEE), AND ANTONIO SERVETTI, (Member, IEEE)**
Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy
Corresponding author: Filippo Gandino (filippo.gandino@polito.it)

**ABSTRACT** In case of an adversary that compromises a node in a wireless sensor network (WSN) and obtains its secret material, techniques for the detection of malicious or compromised nodes have been proposed in several papers. However, the ability to recover secure communications after a compromised node has been detected has not been thoroughly investigated. Such an ability allows to evaluate the level of proper functionality, i.e., the ratio of secure links in the network, that can be recovered after a successful attach and the withdrawal of the compromised secret material. This paper defines and discusses for the first time the recoverability property and provides the formulas to compute it for the main state-of-the-art key management schemes. All the formulas are validated through extensive simulation.

**INDEX TERMS** WSN, key management, recoverability.

## I. INTRODUCTION

Wireless sensor networks (WSNs) are a widely employed pervasive technology. They offer great opportunities for improvement in various fields of human activities (e.g., human behavior recognition [1], healthcare [2] and critical navigation services [3]).

WSNs are typically composed of many low-cost nodes that communicate wirelessly. The nodes are able to sense the environment and to transmit data to a sink. Messages are forwarded through multiple hops from the original sender to the sink. The characteristics of WSNs, especially the low level of resources of the nodes, involve several issues that require specific solutions. The energy constraint requires a careful design of the communication protocols (e.g., MAC [4], routing [5] and cross-layer [6]). There are specific deployment strategies according to the characteristics and application of the network [7]. If the nodes are randomly deployed, specific techniques which detect the position of the nodes can be used [8]. Also security protection requires the design of specific solutions which comply with the characteristics of the network.

The level of security can be a critical factor, according to the importance of a WSN application. Moreover, an adversary could attempt malicious injection of false data in the system or a denial-of-service attack. Eavesdropping is one important issue common to other wireless networks [9]: an adversary could listen, record and, if required, try to decode the messages exchanged by the nodes. Another threat, which is critical for WSNs, is represented by an adversary that compromises one or more nodes of the network. Since the nodes are low-cost, it is normally not possible to use tamper resistant hardware. Therefore, an adversary could obtain all the secret data stored in some nodes. This attack may produce two main consequences: the adversary might impersonate the compromised nodes; the adversary might use the obtained secrets to introduce new malicious nodes to the network and to eavesdrop on some links. Given the high level of risk due to such an attack, many techniques for detecting the presence of malicious nodes have been proposed [10], [11].

The security systems used to protect WSNs are normally based on symmetric cryptography [12]. However, a key management scheme is required for the establishment and distribution of the secret material. In literature, the key management schemes are analyzed in order to evaluate the level of protection provided against the main attacks. In particular, a fundamental property is considered: resilience against an adversary that has compromised some nodes and tries to use the obtained secret material to eavesdrop on other links in the network.

The possibility that a node becomes compromised represents a dangerous threat for a WSN. Specific systems can be used to detect compromised nodes [13]–[15]. Therefore, if malicious nodes are detected, the compromised secret material can be withdrawn. However, if all the secret material shared by two nodes is compromised, they will no more be able to establish safe communications. The analysis of key management schemes normally does not cover the ability to recover secure communications after the secret material withdrawal. The probability of preserving safe communications

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Arafatur Rahman.

with a neighboring node after revoking some compromised secret material, eventually by establishing a new pairwise secret, is here defined as *recoverability*.

In this paper, the recoverability property is defined and discussed. Moreover, the recoverability level of the main state-of-the-art key management schemes is investigated through a comparative analysis. The proposed analytical formulas, which are presented herein for the first time, are validated by extensive simulations. This paper aims to provide a useful support for practitioners interested in existing key management schemes. It can also be of use to researchers for the development and evaluation of future proposals. The recoverability knowledge will provide a deeper understanding of the security characteristics of a network, allowing to correctly select a proper key management scheme. In particular it makes possible to weight the benefits and drawbacks in adopting a key management scheme that favours recoverability versus resilience or vice versa.

The organization of the rest of the paper is as follows: Sect. II presents the background on security in WSNs and key management schemes. In Sect. III-A the recoverability property is defined and a comparative analysis of the recoverability level of the main state-of-the-art schemes is presented. Finally, in Sect. IV some conclusions are drawn.

## II. SECURITY IN WSNS AND KEY MANAGEMENT SCHEMES

The WSNs are composed by low cost devices, so specific security approaches are required. Since in a WSN the messages are transmitted wirelessly, an adversary can eavesdrop on all the traffic, so he/she could know all the data transmitted without encryption. The adversary can inject packets in any point of the network, and he/she is able to replay old messages. Moreover, an adversary could compromise some nodes and obtain all the secret information that are stored by those nodes.

If the malicious nodes, new or compromised, are detected, the secret material shared with those nodes can be withdrawn. However, if the quantity of withdrawn material is too large, the nodes could not be able to recover secure communications.

Since the security in WSNs is normally based on symmetric cryptography, the establishment of common secret keys among the nodes is of fundamental importance. In this section a brief description of the main state-of-the-art key management approaches is presented. For a more in-depth description it is possible to read the existing surveys on key management in WSNs [16], [17].

The current investigation does not consider schemes based on deployment knowledge. Table 1 shows the main characteristics of the analyzed schemes.

### A. FULL PAIRWISE KEYS

In the *Full pairwise keys* (FPWK) [18], a specific pairwise key is shared by each possible couple of nodes independently of their relative distance. Therefore, each node has to store a quantity of keys proportional to the size of the network. The memory required in FPWK corresponds to $(n-1)l_k$, with $l_k$ the size of a key and $n$ the number of nodes in the network. As shown in Table 1, the main drawback of this scheme is represented by the limit to the size of the network, which is due to the memory available to store the secret material. The addition of nodes after the initial deployment of the network is possible. However, each node needs a pairwise key shared with the new node, so the total quantity of nodes must be considered before starting the deployment. Moreover, the size limit due to the memory constraint can never be crossed. FPWK, when it can be applied, provides a high level of resilience against an adversary that has compromised some nodes, since the adversary is able neither to introduce new malicious nodes nor to eavesdrop on a link between two original nodes.

**TABLE 1. Main characteristics of the key management schemes.**

| Schemes | Mobile network | Possibility of node adding | Network size unlimited | Low computational overheads |
|---|---|---|---|---|
| PGK, SKKE, EG & 2C | YES | YES | YES | YES |
| LEAP+ & RSDTMK | NO | YES | YES | YES |
| SSL | YES | YES | YES | NO |
| FPWK, UKP & ST | YES | YES/NO | NO | YES |
| {B-, V- & A-}Blom | YES | YES | YES | NO |

### B. GLOBAL MASTER KEY

In the class of schemes based on a global master key, all the nodes share a master key that is used to establish the final pairwise keys. The basic scheme with a global master key is the *Plain global key* (PGK), where a unique key is used by all the nodes. As shown in Table 1, this approach does not have relevant limitations and its memory and computational overheads are low. In particular, in PGK each node only uses the memory required to store one key ($l_k$). However, this scheme provides a very low resilience against an adversary that has compromised a node, since he/she would be able to eavesdrop on any link and to introduce new malicious nodes that will be authenticated by any original node of the network.

Another important scheme of this class is the *Symmetric-key key establishment* (SKKE). This scheme is adopted by ZigBee.[1] In SKKE, the nodes use a global secret to protect the key establishment. Like PGK, SKKE does not have relevant limitations and it has low memory and computational overheads. In particular, in SKKE a node uses a memory area equal to the size of the global key ($l_k$) plus the size of a node identification (ID) ($l_{ID}$) and of a key ($l_k$) per node within its communication range ($v$). Although the key establishment mechanism used by SKKE improves the security with respect to PGK, the level of resilience against an adversary that has compromised a node is still very low. The adversary would

[1]ZigBee Specification 1.0, June 2005, ZigBee Alliance

be able to introduce new malicious nodes that would pass any authentication check. Moreover, if the adversary has also recorded the messages exchanged during the key establishment, he/she can compute any pairwise key and eavesdrop on any link.

### C. RANDOM KEY PREDISTRIBUTION APPROACHES

The random key predistribution technique is based on the generation of a large quantity of secret material and on the random distribution before the deployment of a part of this material to each node. The main known approach of this class, hereinafter called *EG*, has been proposed by Eschenauer and Gligor [19]. In EG, before the deployment a pool of $p$ keys is generated and a random ring of $r$ keys is assigned to each node. After the deployment, each node looks for shared keys with the other nodes in its communication range. As shown in Table 1, EG does not have special limitations. The memory overhead depends on the size of the ring, since each node stores $r$ keys and $r$ key identification numbers. Moreover, each node stores a key identification number and a node identification number per node in its communication range, in order to match the keys with the nodes. Moreover, each node has to store a node ID per neighboring node ($v$) to identify them and a key ID per neighboring node to know which key must be used during the communications. Therefore, the memory required in EG is $r(l_k + l_{kID}) + v(l_{ID} + l_{kID})$, with $l_{ID}$ the size of a node ID, $l_{kID}$ the size of a key ID and $l_k$ the size of a key. The values of $p$ and $r$ affect the probability of establishing a link between two nodes and the quantity of secret material that an adversary can obtain by compromising a node. The resilience against an adversary that has compromised a node depends on these parameters. A high value of $r$ increases the connectivity, but also increases the memory overhead and decreases the resilience. A high value of $p$ increases the resilience but decreases the connectivity.

An evolution of EG is the *q-composite random key predistribution*(QC) [20]. In this scheme, two nodes have to share at least $q$ *starting keys* in order to establish a link. They compute a hash function on the concatenation of the shared starting keys in order to generate a pairwise key. If a small quantity of nodes is compromised, the level of security is higher than in EG, especially if the value of $q$ is large. According to the analysis provided in [20], $q = 2$ is a good configuration, so this parameter is used for the analysis of this scheme, which hereafter is called 2C.

The main drawback of 2C, with respect to EG, is represented by the higher memory overhead. In EG, the starting keys can be directly used as pairwise keys; in contrast, in QC new pairwise keys are generated and stored. A node has to store $r$ keys and $r$ key IDs. Moreover, each node has to store a node ID per neighboring node to identify them and the pairwise keys shared with each node. Therefore, the memory required in 2C is $r(l_k+l_{kID})+v(l_k+l_{ID})$. By using fixed values for the parameters $p$ and $r$, QC provides a higher resilience, but the memory overhead is higher than in EG. However, by using the same quantity of memory and by guaranteeing the same level of connectivity, QC should use values of $p$ and $r$ that provide a lower level of resilience.

A further evolution of this approach was presented in [21]. A new parameter, $s$ has been added. In this case, $s$ represents the maximum quantity of keys that can be used to generate a pairwise key. The scheme includes a new key establishment routine. The main benefit of this scheme is to reduce the memory overheads. Therefore, with the same memory, it is possible to use a larger ring, and to achieve a better level of resilience. According to the best configuration proposed in the original paper, in the following the values of $q$ and $s$ are set to 1 and 5, respectively. Moreover, the scheme is called 1-5C.

### D. COMBINATORIAL

An approach, based on combinatorial design, is the *Unital-based key predistributed scheme* (UKP) [22]. This scheme is configured according to $t$ and a prime power $m$. The unital design allows to generate blocks of $m + 1$ elements with a fixed probability of having a shared key, since the same key is present in $m^2$ blocks and two blocks cannot share more than one key.

- the pool includes $p = m^3 + 1$ keys,
- the keys are grouped in $m^2\left(m^2 - m + 1\right)$ blocks,
- a block includes $m + 1$ keys,
- the same key is present in $m^2$ blocks,
- the same set composed by more that one key cannot be present in more than one block,
- each node receives $t$ disjointed blocks, so $r = t(m + 1)$ keys,
- the maximum quantity of nodes is $\frac{m^2}{t}\left(m^2 - m + 1\right)$

The authors calculate that $t \sim \sqrt{m}$ is the best configuration. The memory required in UKP is $t(m + 1)(l_k + l_{kID}) + v(l_k + l_{ID})$. As shown in Table 1, the main limitation of UKP is represented by the size limit. This scheme could provide a good level of resilience, but it involves a high memory overhead.

In [23], the authors propose a scheme based on Steiner trades, hereinafter called *ST*. In this case:

- the pool includes $q^2 + q$ keys,
- the key are grouped in $2q^2$ blocks,
- a block includes $k \leq q$ keys,
- the same couple of keys is present in 2 blocks.

This approach does not allow to establish all the direct possible links. Its main benefit is to reduce the memory overhead by reducing the connectivity.

### E. TRANSITORY MASTER KEY

The transitory master key is a global secret that is known by every node, but that is deleted after a time-out. This class of schemes is based on the assumption that an adversary cannot compromise a node in less than a lower bound of time. Therefore, the nodes should be able to use the transitory master secret to establish pairwise keys before an adversary can compromise a node and obtain the master key.

Starting from the deployment to the deletion of the master key a node is within the *initialization phase*. After the deletion of the master key a node is within the *working phase*.

If the time-out for the transitory master key deletion is short, the nodes could not be able to establish the pairwise keys with all their neighboring nodes. However, if the time-out is long, an adversary has more probability to be able of compromising the master key. Normally, an adversary that knows the transitory master key is able to find all the pairwise keys, to decode all the messages and to introduce new malicious nodes.

An important scheme in this class is LEAP+ [24]. It is only compliant with static WSNs. Although LEAP+ provides four kinds of keys, the main scheme is based on the establishment of the pairwise keys. All the nodes know the initial key, which corresponds to the transitory secret, and a keyed pseudo-random function. Each node also has its specific master key, which is computed by executing the pseudo-random function with the initial key on the identifier of that node. After the time-out a node deletes the initial key, but still stores its master key. In order to establish a pairwise key, two nodes within the initialization phase execute the pseudo-random function with the master key of a node on the identifier of the other one. A node within the working phase, which has already deleted the initial key, can only establish pairwise keys with nodes in the initialization phase. The memory required in LEAP+ is $v(l_k + l_{ID}) + l_k$. As shown in Table 1, the main limitation of LEAP+ is represented by the incompatibility with mobile WSNs. However, it has low memory and computational overheads. The level of resilience against an adversary that has compromised a node is very high if the node is compromised in the working phase, but it is very low if the node is compromised in the initialization phase. A new version of this protocol was presented in [25], in order to reduce the key establishment time without compromising the other security properties.

A subsequent scheme in this class is RSDTMK [26]. This scheme merges the transitory master key technique with the random key distribution. In RSDTMK, a pool of $p$ seeds is generated before the deployment. Then, a ring of $r$ seeds per node is randomly selected from the pool. All the nodes know their ring, a keyed pseudo-random function, a simple permutation function and the transitory master key. Two nodes can establish a pairwise key only if they share at least a seed. The nodes execute the permutation function both on a random number between 0 and $2^{\mu}$ and on the shared seed. Then, they execute the keyed pseudo-random function on the previous result. At the end of the initialization, an additional key is generated per unused keys of the ring, in order to be able to establish keys also with nodes deployed later. Therefore, the quantity of $\varrho$ keys stored by a node after the key establishment can be larger than the original ring ($r \leq \varrho \leq r + v - 1$). In the worst case, the same key of the ring is used to generate all the $v$ pairwise key, and one additional key is generated for all the unused $r - 1$ keys. Therefore, each node stores $\varrho$ keys and their IDs, and a node ID and a key ID per neighboring node. The memory required in RSDTMK is $\varrho(l_k + l_{kID}) + v(l_{ID} + l_{kID})$. Like for LEAP+, RSDTMK has low computational overhead and is only compliant with static WSNs. The memory overhead is larger, since the used memory area depends on $r$. If a node is compromised in the working phase, the level of resilience is very high, but slightly lower than LEAP+. However, if the node is compromised in the initialization phase, the level of resilience depends on $p$ and $r$, and it is higher than LEAP+.

### F. PUBLIC CRYPTOGRAPHY

The public cryptography can be used in order to protect the messages used to establish the pairwise keys. Many cryptographic primitives can be used in order to limit the computational effort without compromising security [27]. In [28], the authors proposed a scheme based on the Secure Sockets Layer (SSL) handshake.[2] In the following this approach is called SSL. In SSL, each node stores a specific couple of public and secret keys, a certificate that guarantees its authenticity and the public key of the administrator. The certificate of a node is composed by its public key, its identification number and a signature computed over the certificate by the administrator with its secret key. A node verifies the authenticity of the certificates by using the public key of the administrator and encrypts the messages sent during the handshake with the public key of the receiver. The memory required in SSL is $v(l_k + l_{ID}) + l_{pr} + 2l_{pu} + l_{sign}$, with $l_{pr}$ the length of a private, $l_{pu}$ the length of a public key, and $l_{sign}$ the length of a signature. As shown in Table 1, SSL involves a relevant computational overhead, since the nodes have to compute modular exponential operations. However, SSL provides a high level of resilience against an adversary that has compromised some nodes.

### G. K-SECURE

The $k$-secure family of protocols is based on Blom's scheme. A symmetric matrix over a finite field with $k$ columns and rows is used to generate the secrets of the nodes. Each node has a public information corresponding to a vector of $k$ numbers. Before deployment, a secret per node is generated by multiplying its vector by the matrix. The resulting vector of $k$ numbers is stored by the node. Two nodes can generate a shared key by exchanging their public vector, and by multiplying their own secret vector by the public one of the other node. The result is a common secret. This approach is $k$-secure, since an adversary that obtains $k$ secret vectors can find the original matrix and compromise the whole security.

This technique has been applied by many approaches. In [29], hereinafter called basic Blom (*B-Blom*), it was used to generate end-to-end keys and group keys. A drawback, common to all the approaches based on Blom's scheme, is represented by the computational overhead. The memory required by the secret material before deployment is $(2k + 1) \cdot l_k$, with $l_k$ the length of a number that can be equivalent to a

[2]The SSL Protocol Version 3.0. http://home.netscape.com/eng/ssl3/

pairwise key. After deployment, the generated keys and the corresponding node identifiers must also be stored. In [30], the public vector is part of a Vandermonde matrix. The advantage is that each node has to store and exchange only the first number of the vector, since the other one can be generated starting from it. The drawback is that three modular multiplications per number are required, while B-Blom requires only one modular multiplication. Hereinafter this approach is called Vandermonde Blom (*V-Blom*) In [31], an evolution, hereinafter called advanced Blom (*A-Blom*), of that protocol was proposed. Each node is matched to one of $c$ classes and can communicate only with the nodes matched with the other ones. An adversary needs $k$ secret vectors of the same class to compromise the network.

### H. RECOVERABILITY AND KEY REVOKING IN LITERATURE

Previous works on key management for WSNs often do not consider the recoverability problem, and in particular they do not present a mathematical evaluation of this property. In the analysis of EG [19], the authors state that by removing the secret material of a compromised node some links may disappear, but only few nodes may be affected. In the analysis of q-composite [20], the authors state that their approach can support node revocation, but they do not evaluate its effects. For LEAP+ [24], the authors conclude that there are no problems after node revoking. However, according to their assumptions, they do not consider neither the recoverability nor the revoking if a node is compromised during the initialization. In other papers, neither the recoverability nor the revoking are considered (e.g., UKP [22], RSDTMK [26], SSL [28]). In literature, there are also many surveys of key management schemes for WSNs. Among them, the best contribution related to recoverability has been presented in [18]. This paper considers the effects of revoking keys and proposes a rating for existing schemes from very easy to very difficult. However, a quantitative evaluation and the relation with the quantity of compromised nodes are not considered. In [17], [32] and [33], the revocation technique is analyzed, but the recoverability is not discussed. In other surveys, neither the recoverability nor the revoking are considered (e.g., [16], [34]).

## III. RECOVERABILITY DEFINITION AND COMPARATIVE ANALYSIS

This section presents the definition of recoverability and analyzes the level of connectivity and recoverability provided by the considered key management schemes. The formulas of the connectivity have been partially presented in the original papers, partially in previous studies [26]. The formulas of the recoverability are presented in this paper for the first time.

### A. RECOVERABILITY DEFINITION

A WSN can adopt various malicious node detection techniques. The goal of these techniques is to identify compromised nodes. After a compromised node is detected, all its secret material and subsequent secrets based on the same material are revoked, so messages are broadcasted requiring that all the nodes delete this material. Recoverability is the probability that two nodes can still establish a link after some compromised secret material was withdrawn, with maximum equal to one and minimum equal to zero. Therefore, the recoverability can be considered as the connectivity after revoking some secret material, i.e., the probability that two generic neighboring nodes have established a secret and that secret has not been revoked plus the probability that the secret has been revoked but it is possible to establish a new one.

Recoverability is also connected to the resilience. Resilience against eavesdropping can be computed as the ratio of links that cannot be eavesdropped by an adversary that has already compromised some nodes. The maximum level of resilience is one and the minimum is zero. If an adversary that has compromised some nodes has obtained the possibility to eavesdrop on a large part of the links, it is probable that after revoking the compromised material there will be few safe links. However, recoverability and resilience are not directly proportional. In fact, with respect to a scheme that provides a high level of resilience but a low connectivity, a scheme that allows the nodes to easily establish new links could have a lower resilience, but after revoking some secret material it will be able to recover a larger part of the links.

Resilience represents the security level of the network between the moment in which an opponent has compromised some nodes and the moment in which the compromised nodes are identified. Recoverability represents the level of operability of the network after the identification of the compromised nodes and the withdrawn of the compromised material. Therefore, the administrator of a WSN can use the recoverability to make a better selection of the proper key management scheme. If two protocols have a similar level of resilience, the level of recoverability can be an important discriminating factor. Moreover, if the adopted malicious node detection technique is considered reliable, recoverability is even more important than resilience, since the window of time in which the resilience is relevant is short.

### B. CONNECTIVITY

The existing key management schemes do not always allow a node to establish a link with any neighboring node. In order to analyze the number of links that are recovered after the withdrawal of some secret material, it is required to consider their initial number. The probability to establish a link with a neighboring node is defined as connectivity. Therefore, the level of connectivity also represents the average fraction of possible links that are actually established.

FPWK, SSL, SKKE, PGK, B-Blom and V-Blom provide a optimum level of connectivity, since all the possible links are always established. Even LEAP+ provides the same level of connectivity, but only if the time-out before the deletion of the transitory secret is not too short. Otherwise, the nodes would not be able to complete the key establishment. The connectivity level of EG, 2C, 1-5C and RSDTMK depends on $r$ and $p$. EG, 1-5C and RSDTMK provide a level of connectivity

**TABLE 2.** Recoverability for mobile scheme.

| Scheme | Recoverability, $x$ nodes compromised |
|---|---|
| FPWK & SSL | 1 |
| PGK & SKKE | 0 |
| EG & 1-5C | $\sum_{i=1}^{r} \frac{\binom{r}{i}\binom{p-r}{r-i}\sum_{j=1}^{i}\binom{i}{j}(-1)^{j+1}\binom{p-j}{r}^{x}}{\binom{p}{r}^{x+1}}$ |
| 2C | $\sum_{i=2}^{r} \frac{\binom{r}{i}\binom{p-r}{r-i}\sum_{j=2}^{i}\binom{i}{j}(-1)^{j}(j-1)\binom{p-j}{r}^{x}}{\binom{p}{r}^{x+1}}$ |
| UKP | $1-\left(1-\frac{(m+1)^2}{m^3+m+1}\left(1-\frac{m+1}{m^3+1}\right)^{xt}\right)^{t^2}$ |
| B-Blom & F-Blom | 1 , for $x < k$<br>0 , for $x \geq k$ |

equal to $1-\binom{p-r}{r}/\binom{p}{r}$; in 2C is $1-\left(\binom{p-r}{r}+\binom{r}{1}\binom{p-r}{r-1}\right)/\binom{p}{r}$. The connectivity is higher with high values of $r$ and low values of $p$, since each node has a large part of the keys/seeds used in the network. However, a high value of $r$ requires a large memory area to store the ring. Moreover, the level of resilience is higher with high values of $p$ and low values of $r$, since an adversary that has compromised some nodes knows a small part of the keys/seeds in the network. It is observed that with the same parameters 2C provides a lower level of connectivity. In UKP the level of connectivity is equal to $1-\left(1-\frac{(m+1)^2}{m^3+m+1}\right)^{t^2}$. A high value of $m$ decreases the level of connectivity, while a high value of $t$ increases it. A-Blom provides a connectivity level dependent on the number of classes: $\frac{l-1}{l}$. ST has a level of connectivity equal to $\frac{k(k-1)}{2(2q^2-1)}$. Since $k \leq q$, the maximum level of connectivity is about 0.25.

### C. RECOVERABILITY ANALYSIS

Table 2 shows the formulas for the recoverability of mobile approaches, while Table 3 shows the formulas for static approaches. FPWK and SSL provide a optimum level of recoverability, since they also have an optimum level of resilience. Even LEAP+ provides a optimum level of recoverability, but only if the nodes are compromised after the deletion of the transitory master secret. Otherwise, LEAP+ provides no recoverability, since an adversary with the initial key can introduce new malicious nodes and potentially eavesdrop on all the links in the WSN. Neither PGK nor SKKE provide any recoverability, since they also have a low level of resilience.

For schemes that exploit the random distribution of secret material, the level of recoverability provided depends on their parameters. The level of recoverability with EG is equal to the probability that two nodes share some keys multiplied by the probability that at least one of these keys is not compromised. The probability that two nodes share a number of keys $i$ between 1 and $r$ is $\sum_{i=1}^{r}\frac{\binom{r}{i}\binom{p-r}{r-i}}{\binom{p}{r}}$. If we consider $i$ shared keys, the probability that all the keys from the $x$ compromised rings do not include one of the $i$ keys is $\frac{\binom{i}{1}\binom{p-1}{r}^{x}}{\binom{p}{r}^{x}}$. However, this equation for probability double-counts the cases in which two of the $i$ keys are not compromised. To correct the formula, it is necessary to subtract the probability that two keys are not compromised: $-\frac{\binom{i}{2}\binom{p-2}{r}^{x}}{\binom{p}{r}^{x}}$. However, the possibility that three keys are not compromised has been subtracted so that it is now not considered by the formula. The correct formula is obtained by alternately adding and subtracting the probabilities that $j$ keys are not compromised, with $j$ from 1 to $i$. With the same parameters, 1-5C provides the same level of recoverability as EG, since the connection between two nodes is based on the same assumptions.

The formula of the recoverability for 2C is very similar to the same formula for EG. The differences are that: (i) the general summation starts from 2, since two nodes can establish a pairwise key only if they share at least 2 keys; (ii) the second summation also starts from 2, since a link can be recovered only if at least 2 keys are not compromised; (iii) since the sequence of addition and subtraction of the second summation starts from 2, the formula used to correct the redundant sets of not compromised keys requires a factor equal to the quantity of keys in the set minus one: $(j-1)$.

The formula of recoverability for UKP is very similar to the connectivity formula. For the recoverability, the probability that two blocks share a key, $\frac{(m+1)^2}{m^3+m+1}$, is multiplied by the probability that this specific key is not included in the $xt$ compromised blocks: $\left(1-\frac{m+1}{m^3+1}\right)^{xt}$.

B-Blom and F-Blom, which are k-secure, provide a full recoverability if the number of compromised nodes is lower than $k$, otherwise the recoverability level is zero, since the secret matrix is compromised and the adversary can generate all the keys.

The formula of recoverability for RSDTMK, if the nodes are compromised during the initialization phase, corresponds to the probability that two nodes share at least one seed, $1-\frac{\binom{p-r}{r}}{\binom{p}{r}}$, multiplied by the probability that $i$ other keys are randomly the same, $\sum_{i=0}^{\varrho-1}\frac{\binom{\varrho-1}{i}\binom{2^{\mu}p-\varrho}{\varrho-i-1}}{\binom{2^{\mu}p-1}{\varrho-1}}$, multiplied by the probability that at least one of the $i+1$ shared keys is not included in the $x$ compromised rings of $r$ seeds: $\sum_{j=1}^{i+1}\frac{\binom{i}{j}(-1)^{j+1}\binom{2^{\mu}p-j}{2^{\mu}r}^{x}}{\binom{2^{\mu}p}{2^{\mu}r}^{x}}$. However, this formula includes a marginal approximation, since the $\varrho$ keys are considered independent from the starting $r$ seeds.

The formula of recoverability for RSDTMK, if the nodes are compromised during the working phase, is similar to the previous one. It corresponds to the probability that two nodes share at least one seed, $1-\frac{\binom{p-r}{r}}{\binom{p}{r}}$, multiplied by the probability that $i$ other keys are randomly the same, $\sum_{i=0}^{\varrho-1}\frac{\binom{\varrho-1}{i}\binom{2^{\mu}p-\varrho}{\varrho-i-1}}{\binom{2^{\mu}p-1}{\varrho-1}}$, multiplied by the probability that at least one of the $i+1$ shared keys is not included in the $x$ compromised rings of $\varrho$ keys: $\sum_{j=1}^{i+1}\frac{\binom{i}{j}(-1)^{j+1}\binom{2^{\mu}p-j}{\varrho}^{x}}{\binom{2^{\mu}p}{\varrho}^{x}}$. Even this formula includes

**TABLE 3.** Recoverability for static schemes.

| Scheme | Recoverability, $x$ nodes compromised in the working phase | Recoverability, $x$ nodes compromised in the initialization phase |
|---|---|---|
| LEAP+ | 1 | 0 |
| RSDTMK | $\left(1-\frac{\binom{p-r}{r}}{\binom{p}{r}}\right)\sum_{i=0}^{\varrho-1}\frac{\binom{\varrho-1}{i}\binom{2^{\mu}p-\varrho}{\varrho-i-1}\sum_{j=1}^{i+1}\binom{i}{j}(-1)^{j+1}\binom{2^{\mu}p-j}{\varrho}^{x}}{\binom{2^{\mu}p-1}{\varrho-1}\binom{2^{\mu}p}{\varrho}^{x}}$ | $\left(1-\frac{\binom{p-r}{r}}{\binom{p}{r}}\right)\sum_{i=0}^{\varrho-1}\frac{\binom{\varrho-1}{i}\binom{2^{\mu}p-\varrho}{\varrho-i-1}\sum_{j=1}^{i+1}\binom{i}{j}(-1)^{j+1}\binom{2^{\mu}p-j}{2^{\mu}r}^{x}}{\binom{2^{\mu}p-1}{\varrho-1}\binom{2^{\mu}p}{2^{\mu}r}^{x}}$ |

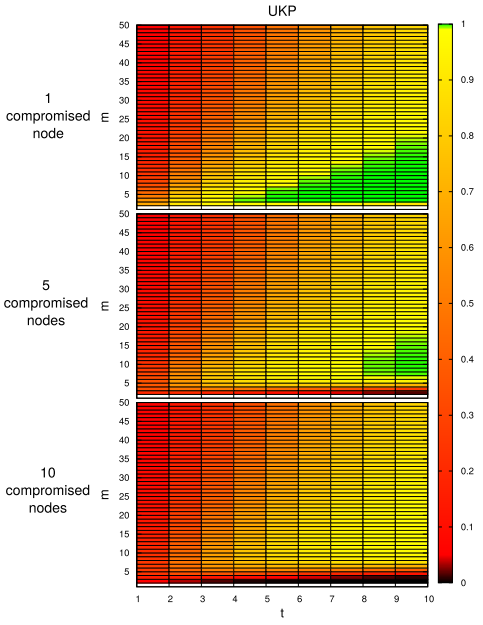


**FIGURE 1.** Recoverability of UKP.

a marginal approximation, since the $\varrho$ keys are considered independent from the starting $r$ seeds.

Fig. 1 shows the level of recoverability for UKP, which is based on $m$ and $t$, while Fig. 2 shows the level of recoverability for the schemes that depend on $r$ and $p$. The green colour represents a recoverability level higher than 0.99. Then, from 0.99 to 0.05 the color changes from yellow to red. Finally, from 0.05 to 0 the colour changes from red to black. In EG, the main effect of nodes being compromised is to reduce the level of recoverability especially if $r$ is close to $p$, since the adversary has the majority of the keys. In 2C, the level of recoverability, like the level of connectivity, is generally lower than in EG. The compromised nodes generate a strong decrease in the level of recoverability both if $r$ is close to $p$ and if $r$ is low, since in this case the withdrawal of some keys prevents the possibility that the nodes still share two keys. In RSDTMK, the main effects of the nodes being compromised during the working phase consists in a slight decrease in the level of recoverability if $r$ is close to $p$. However, here the level of recoverability is generally better than in EG. The main effects of the nodes being compromised during the initialization is a strong decrease in the level of recoverability if $r$ is close to $p$. In UKP, the main effect of the nodes being compromised is to reduce the level of recoverability especially if $t$ is close to $m$.

The level of recoverability depends on the key management scheme adopted, which is selected taking into account the network architecture. However, two networks with the same key management scheme but different architectures will have the same recoverability. For example, let's consider two WSNs. The first one uses EG, the key management scheme, and Flooding [35], a data centric architecture in which each sensor forwards any message until the destination or a maximum number of hops are reached. The second one uses EG and LEACH [36], a hierarchical architecture in which: the nodes are grouped in clusters; in each cluster there is a head; a head establishes one link per node in its cluster; a head establishes one link per cluster head in its communication range; the head randomly rotates among the nodes in the cluster. In both WSNs the recoverability level is correctly described by the proposed formulas, since the distribution of the rings of keys is not affected by the architecture.

### D. COMPARISON

According to the results shown in Fig. 1 and 2, it is not possible to establish a general rule to determine the optimal configuration, since a different quantity of compromised nodes corresponds to a different optimal configuration. In order to obtain a quantitative comparison of the schemes, the following methodology has been used. A network with a number of nodes $n = 500$, in which each node has a number of neighbors $v = 10$ is considered. The key length ($l_k$) is set to 128 bits, the node IDs length($l_{ID}$) to 16 bits and the key IDs ($l_{kID}$) to 8 bits. The length of the private key ($l_{pr}$), of the public key ($l_{pu}$), of the signature ($l_{sign}$) in SSL are set to 512 bits. For 2C, 1-5C, RSDTMK and EG the values for $r$ and $p$ (for UKP, $m$ and $t$) have been selected so that the memory storage is less than 5% of the RAM memory of Tmotes Sky (512 bytes). The memory threshold corresponds to an upper bound for $r$ and for the value of $t(m+1)$. In EG, $r \leq 28$, in 2C, $r \leq 19$, in 1-5C, $r \leq 25$, in RSDTMK, $r \leq 24$, in UKP, $t(m+1) \leq \frac{180}{17}$. The memory formulas are described in Section II.

A connectivity higher than or equal to 0.99 has been selected as a constraint for the selection of the suitable parameters. This threshold, with a set value of $r$, corresponds to an upper bound for $p$.

A high value of $\mu$ in RSDTMK improves the resilience, but it increases the memory storage. In order to provide a high level of resilience with limited drawbacks, $\mu$ is set to 8.

According to the connectivity and memory size constraints of the proposed study case and to the characteristics of UKP presented in Section II-C, there does not exist a configuration

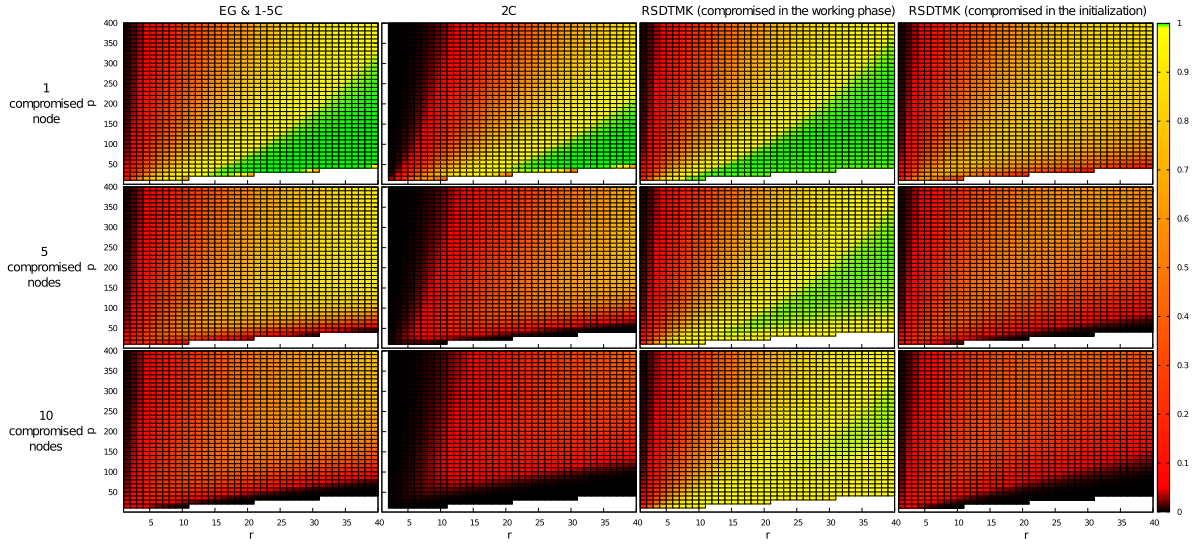


**FIGURE 2.** Recoverability of EG, 2C and RSDTMK.

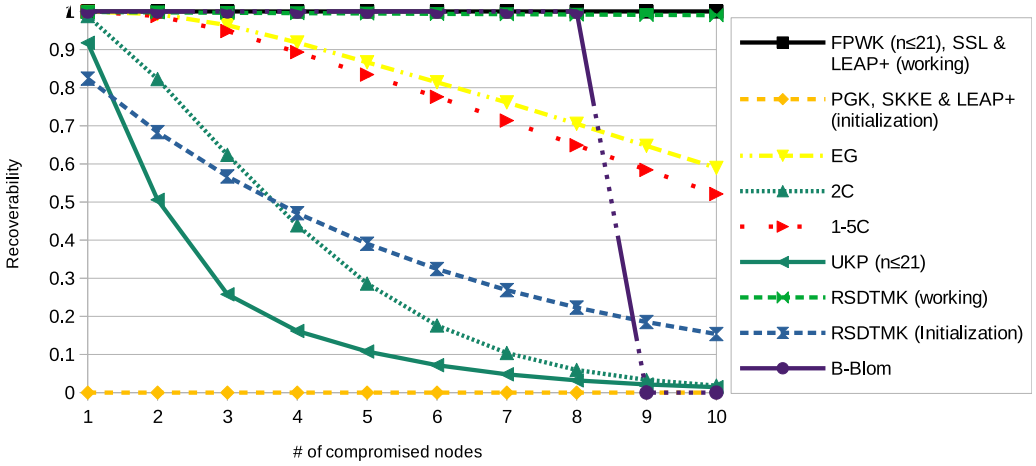


**FIGURE 3.** Recoverability, according to the number of compromised nodes, with memory limit 512 bytes, and connectivity larger than 0.99.

of UKP compliant with 500 nodes, since the largest possible network corresponds to $n = 21$, $m = 3$ and $t = 3$. Even FPWK is not complaint with a network composed by 500 nodes. The maximum quantity of nodes compliant with the memory upper bound is $n = 31$.

The level of A-Blom depends on the quantity of classes. However, the memory overhead is proportional to the number of classes. Since 100 classes are required to provide a connectivity equal to 0.99, this protocol is not complaint with the memory requirement and it is not included in the comparison. B-Blom can be implemented with $k = 9$, while V-Blom with $k = 18$.

Since the level of connectivity of ST is no higher than 0.25, that scheme is not included in the comparison.

Fig. 3 shows the level of recoverability provided by the analyzed schemes. For all the considered quantity of compromised nodes, all the schemes have been tested with all the configurations complaint with the connectivity and memory thresholds, and in each case the higher level of recoverability has been selected. For all the cases, the largest $r$ corresponds to the higher level of recoverability. In contrast, the value of $p$ that provides the higher level of recoverability is not the same for all the possible numbers of compromised nodes. For a large quantity of compromised nodes the highest $p$ provides the best recoverability, but with a low quantity of compromised nodes a lower $p$ is better. V-Blom has the same recoverability of B-Blom, but with a higher value of $k$. Therefore, it is not included in the chart, in order to provide a more clear and focused comparison.

It is possible to observe that, with the same memory overhead and by guaranteeing the same connectivity level, FPWK and SSL provide full recoverability, since all the links among the not compromised nodes are safe. However, FPWK can be applied only to networks with at most 21 nodes, while SSL

has the largest computational overheads among the analyzed schemes. Even LEAP+ provides full recoverability, but only if the adversary is not able to compromise nodes before its timeout. Otherwise, the level of recoverability provided by LEAP+ is zero. Moreover, LEAP+ can only be applied to static networks. Even RSDTMK provides a level of recoverability close to the optimum only if the adversary is not able to compromise nodes before its timeout. If the nodes are compromised before the timeout, RSDTMK is still able to provide a good level of recoverability. However, even RSDTMK can be applied only to static networks. B-Blom and V-Blom provide an optimal level of recoverability if $x$, the number of compromised nodes, is lower than $k$. V-Blom provides a recoverability equal to 1 for $x < 18$, but it requires 27 modular multiplications per node per pairwise key, while B-Blom requires only 9 modular multiplications. If $x \geq k$, EG provides the best recoverability among the schemes that do not have strict requirements, otherwise B-Blom and V-Blom are better. 2C provides a lower level of recoverability than EG. However, as described in literature [20], [26], for a low number of compromised nodes, 2C provides better resilience than EG. This difference is due to the fact that 2C has a lower probability of recovering a link that has been revoked, since it needs 2 shared keys, while EG requires only one key. In 1-5C, each node stores more keys than in 2C, but less than in EG. The larger number of keys allows 1-5C to provide a better level of resilience than 2C. Although the number of keys is lower than in EG, the opportunity to use all the shared keys to compute a pairwise key provides a better level of resilience also with respect to EG. However, the recoverability in 1-5C and in EG is based on the same formula, so EG reaches a better level, thanks to the higher quantity of keys. UKP provides a low recoverability and can be applied only to small networks. The worst level of recoverability is provided by PGK and SKKE, since all the links are compromised.

The comparative results provided by an analysis based either on recoverability or on resilience are not always the same. For example it is possible to observe EG and 1-5C. They have the same requirements and they can be used in the same scenarios. EG provides a slightly higher level of recoverability than 1-5C. However, according to the analysis provided in [21], 1-5C has a higher level of resilience. In this case, the administrator of the network has to carefully balance which protocol is the best for its network.

### E. VALIDATION

In order to validate the proposed formulas, the considered schemes have been simulated and the results have been compared with the results obtained by the analytical evaluation. In particular the key distribution of EG, 2C and RSDTMK have been implemented by a simulator written in C language. Per each scheme, 3 scenarios with a 1, 5 and 10 compromised nodes, and 25 configurations per scenario, with $1 \leq p \leq 401$ and $1 \leq r \leq 41$, have been executed $10^6$ times. In the simulations, the sets of keys known by the nodes are generated following the rules of the specific scheme. Then, some sets are compromised, and the recoverability between two nodes is checked. For example, in order to simulate EG, a set of $r$ different random numbers between 0 and $p$ per compromised node is extracted. Then, 2 sets of $r$ numbers are extracted. The last two sets represent the rings of two valid nodes. All the compromised keys are deleted from the valid sets. The link is recovered only if the two sets still share at least one key.

**TABLE 4. Difference between the analytical results and the simulations in the recoverability level of EG.**

| Compromised nodes | Analytical | Simulations | Difference |
|---|---|---|---|
| 1 | 0.999970 | 0.999970 | 0.000000 |
| 2 | 0.993028 | 0.993010 | 0.000018 |
| 3 | 0.964088 | 0.964083 | 0.000005 |
| 4 | 0.918571 | 0.918526 | 0.000045 |
| 5 | 0.866605 | 0.866637 | 0.000032 |
| 6 | 0.814326 | 0.814291 | 0.000035 |
| 7 | 0.761519 | 0.761609 | 0.000090 |
| 8 | 0.705366 | 0.705347 | 0.000019 |
| 9 | 0.647547 | 0.647575 | 0.000028 |
| 10 | 0.589632 | 0.589642 | 0.000010 |

The difference between the simulative result and the analytical result for EG and 2C was always less than $10^{-3}$. In the majority of the configurations the difference for RSDTMK was also lower than $10^{-3}$, but for some configuration with $r$ very close to $p$ the difference was larger (always less than $10^{-2}$). This larger error is due to the approximation introduced in the formulas for RSDTMK. However, these configurations do not provide a high level of recoverability, so they are not relevant for the analysis of the scheme. Therefore, it is observed that these errors do not introduce visible modifications in the comparative analysis provided in the previous section. In order to provide a clear idea of the difference between the results obtained by using the proposed formulas and the results of the simulations, Table 4 shows the values of recoverability for EG with the configuration used in the comparison.

## IV. CONCLUSION

In this paper, the recoverability property, which represents a fundamental feature of security systems for WSNs, has been defined and discussed. The state-of-the-art key management schemes for WSNs have been investigated in order to evaluate their ability to recover secure communications after the secret material owned by some nodes has been withdrawn. The correctness of the presented analytical formulas has been validated by simulations.

Taking into account the presented analysis, when the main security goal is to provide the best recoverability (i.e., to guarantee the largest number of secure links after some compromised nodes are detected and their compromised secret material is withdrawn), according to the characteristics of the network the best scheme is:

- FPWK, if the number of node is very low;
- LEAP+, if the network is static and it is assumed that the adversaries are not able to compromise a node before the time required to complete the key establishment;
- SSL, if the high computational overhead is not a constraint for the network;
- V-Blom, if the computational overhead of this scheme is compliant with the network constraints and a higher security for $x < k$ is considered better than a higher security for $x \geq k$.
- B-Blom, if the computational overhead of this scheme is compliant with the network constraints and a higher security for $x < k$ is considered better than a higher security for $x \geq k$.
- EG.

This study represents a valuable aid for the analysis of existing key management schemes and it can be used as a further evaluation metric for the design of future security systems for WSNs. The proposed formulas can be used after compromised nodes are detected, in order to calculate how many links per node there will be in the network (by multiplying the current quantity of links by the recoverability level). Therefore, they can be used to evaluate if withdrawing all the nodes is better or worse than revoking the compromised keys.

## REFERENCES

[1] D. Tao, L. Jin, Y. Wang, and X. Li, "Rank preserving discriminant analysis for human behavior recognition on wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 813–823, Feb. 2014.
[2] G. Loubet, A. Takacs, and D. Dragomirescu, "Implementation of a battery-free wireless sensor for cyber-physical systems dedicated to structural health monitoring applications," *IEEE Access*, vol. 7, pp. 24679–24690, 2019.
[3] C. Wang, H. Lin, R. Zhang, and H. Jiang, "SEND: A situation-aware emergency navigation algorithm with sensor networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 1149–1162, Apr. 2017.
[4] E. Egea-López, J. Vales-Alonso, A. Martínez-Sala, J. García-Haro, P. Pavón-Mariño, and M. V. B. Delgado, "A wireless sensor networks MAC protocol for real-time applications," *Pers. Ubiquitous Comput.*, vol. 12, no. 2, pp. 111–122, 2008.
[5] X. Shao, C. Wang, C. Zhao, and J. Gao, "Traffic shaped network coding aware routing for wireless sensor networks," *IEEE Access*, vol. 6, pp. 71767–71782, 2018.
[6] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. Hanzo, "Cross-layer network lifetime optimisation considering transmit and signal processing power in wireless sensor networks," *IET Wireless Sensor Syst.*, vol. 4, no. 4, pp. 176–182, Dec. 2014.
[7] V. N. Vo, T. G. Nguyen, C. So-In, and D.-B. Ha, "Secrecy performance analysis of energy harvesting wireless sensor networks with a friendly jammer," *IEEE Access*, vol. 5, pp. 25196–25206, Oct. 2017.
[8] T. Hazim, G. Karagiannidis, and T. A. Tsiftsis, "Probability of early detection of ultra-wideband positioning sensor networks," *IET Wireless Sensor Syst.*, vol. 1, no. 3, pp. 123–128, Sep. 2011.
[9] Y. Zou, X. Wang, and W. Shen, "Eavesdropping attack in collaborative wireless networks: Security protocols and intercept behavior," in *Proc. IEEE 17th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, Jun. 2013, pp. 704–709.
[10] W. R. Pires, T. H. de P. Figueiredo, H. C. Wong, and A. A. F. Loureiro, "Malicious node detection in wireless sensor networks," in *Proc. 18th Int. Parallel Distrib. Process. Symp.*, Apr. 2004, p. 24.
[11] S. H. Jokhio, I. A. Jokhio, and A. H. Kemp, "Node capture attack detection and defence in wireless sensor networks," *IET Wireless Sensor Syst.*, vol. 2, no. 3, pp. 161–169, Sep. 2012.
[12] Y. Zhou, Y. Fang, and Y. Zhang, "Securing wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 10, no. 3, pp. 6–28, Third 2008.
[13] H. Moosavi and F. M. Bui, "A game-theoretic framework for robust optimal intrusion detection in wireless sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 9, pp. 1367–1379, Sep. 2014.
[14] S. Shamshirband, N. B. Anuar, M. L. M. Kiah, V. A. Rohani, D. Petković, S. Misra, and A. N. Khan, "Co-FAIS: Cooperative fuzzy artificial immune system for detecting intrusion in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 42, pp. 102–117, Jun. 2014.
[15] C.-M. Yu, Y.-T. Tsou, C.-S. Lu, and S.-Y. Kuo, "Localized algorithms for detection of node replication attacks in mobile sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 5, pp. 754–768, May 2013.
[16] J. C. Lee, V. C. M. Leung, K. H. Wong, J. Cao, and H. C. B. Chan, "Key management issues in wireless sensor networks: Current proposals and future developments," *IEEE Wireless Commun.*, vol. 14, no. 5, pp. 76–84, Oct. 2007.
[17] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway, "A survey of key management schemes in wireless sensor networks," *Comput. Commun.*, vol. 30, nos. 11–12, pp. 2314–2341, Sep. 2007.
[18] M. A. Simplício, Jr., P. S. L. M. Barreto, C. B. Margi, and T. C. M. B. Carvalho, "A survey on key management mechanisms for distributed wireless sensor networks," *Comput. Netw.*, vol. 54, no. 15, pp. 2591–2612, 2010.
[19] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. 9th ACM conf. Comput. Commun. Secur. (CCS)*, Nov. 2002, pp. 41–47.
[20] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. Symp. Secur. Privacy*, May 2003, pp. 197–213.
[21] F. Gandino, R. Ferrero, and M. Rebaudengo, "A key distribution scheme for mobile wireless sensor networks: $q$ - $s$ -composite," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1, pp. 34–47, Jan. 2017.
[22] W. Bechkit, Y. Challal, A. Bouabdallah, and V. Tarokh, "A highly scalable key pre-distribution scheme for wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 2, pp. 948–959, Feb. 2013.
[23] S. Ruj, A. Nayak, and I. Stojmenovic, "Pairwise and triple key distribution in wireless sensor networks with applications," *IEEE Trans. Comput.*, vol. 62, no. 11, pp. 2224–2237, Nov. 2013.
[24] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks," *ACM Trans. Sensor Netw.*, vol. 2, no. 4, pp. 500–528, 2006.
[25] F. Gandino, R. Ferrero, B. Montrucchio, and M. Rebaudengo, "Fast hierarchical key management scheme with transitory master key for wireless sensor networks," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1334–1345, Dec. 2016.
[26] F. Gandino, B. Montrucchio, and M. Rebaudengo, "Key management for static wireless sensor networks with node adding," *IEEE Trans Ind. Informat.*, vol. 10, no. 2, pp. 1133–1143, May 2014.
[27] K.-A. Shim, "A survey of public-key cryptographic primitives in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 577–601, 1st Quart., 2016.
[28] A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in *Proc. 3rd IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2005, pp. 324–328.
[29] L. Harn, C.-F. Hsu, O. Ruan, and M.-Y. Zhang, "Novel design of secure end-to-end routing protocol in wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 6, pp. 1779–1785, Mar. 2016.
[30] M. Rahman and S. Sampalli, "An efficient pairwise and group key management protocol for wireless sensor network," *Wireless Pers. Commun.*, vol. 84, no. 3, pp. 2035–2053, Oct. 2015.
[31] A. Albakri and L. Harn, "Non-interactive group key pre-distribution scheme (GKPS) for end-to-end routing in wireless sensor networks," *IEEE Access*, vol. 7, pp. 31615–31623, 2019.
[32] J. Zhang and V. Varadharajan, "Wireless sensor network key management survey and taxonomy," *J. Netw. Comput. Appl.*, vol. 33, no. 2, pp. 63–75, 2010.
[33] X. He, M. Niedermeier, and H. de Meer, "Dynamic key management in wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 36, no. 2, pp. 611–622, 2013.
[34] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A survey," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 2, pp. 52–73, 2nd Quart., 2009.
[35] A. S. Tanenbaum and D. J. Wetherall, *Computer Networks*, 5th ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2011.
[36] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

**FILIPPO GANDINO** received the M.S. and Ph.D. degrees in computer engineering from the Politecnico di Torino, in 2005 and 2010, respectively. He is currently an Assistant Professor with the Dipartimento di Automatica e Informatica, Politecnico di Torino. His research interests include ubiquitous computing, RFID, WSNs, security and privacy, network modeling, and quantum computing.

**ANTONIO SERVETTI** received the M.S. degree in computer engineering, and the Ph.D. degree in computer engineering from the Politecnico of Turin, Italy, in 1999 and 2004, respectively. In 2003, he was a Visiting Scholar, supervised by Prof. J. D. Gibson, at the Signal Compression Laboratory, University of California, Santa Barbara, CA, USA, where he worked on selective encryption for speech transmission over packet networks. He is currently an Assistant Professor on computer engineering with the Dipartimento di Automatica ed Informatica, Politecnico di Torino. His research focuses on multimedia processing and encryption, multimedia communications over wired and wireless packet networks, and real-time multimedia network protocols.