Scientia Iranica D (2016) 23(6), 2831–2844

Sharif University of Technology
Scientia Iranica
Transactions D: Computer Science & Engineering and Electrical Engineering
www.scientiairanica.com

Invited/Review Article

A survey of key pre-distribution and overlay routing in unstructured wireless networks

M. Gharib[a], **H. Yousefi'zadeh**[b,*] **and A. Movaghar**[c]

a. *Department of Computer Science, Institute for Research in Fundamental Sciences, Tehran, Iran.*
b. *Center for Pervasive Communications and Computing, University of California, Irvine, USA.*
c. *Department of Computer Engineering, Sharif University of Technology, Tehran, Iran.*

Received 4 April 2016; received in revised form 26 April 2016; accepted 17 June 2016

KEYWORDS
Security;
Wireless networks;
Cryptography;
Key pre-distribution;
Overlay routing.

Abstract. Unstructured wireless networks such as mobile ad hoc networks and wireless sensor networks have been rapidly growing in the past decade. Security is known as a challenging issue in such networks, in which there is no fixed infrastructure or central trusted authority. Further, node limitations in processing power, storage, and energy consumption add further complexity to addressing security in such networks. While cryptography has proven to be an effective solution capable of satisfying most network security requirements, it requires the use of efficient key pre-distribution algorithms compatible with the limitation of unstructured wireless networks. Typically, a key pre-distribution algorithm forms a cryptographic overlay layer above the network routing layer and as such introduces the need for relying on two layers of routing for secure delivery of information. In this paper, we conduct a categorical review of key pre-distribution methods for unstructured wireless networks. We also compare different key pre-distribution schemes in terms of performance and security strength. Finally, we provide an overview of recent overlay routing algorithms relying on key pre-distribution.

1. Introduction

Unstructured wireless networks have attracted the attention of many researchers as the result of explosive growth in wireless technology. Mobile ad hoc networks (MANETs) and Wireless Sensor Networks (WSNs) are considered as the most popular kinds of such networks. Vehicular ad hoc networks (VANETs), Wireless Mesh Networks (WMNs), and Smart Phone Ad hoc Networks (SPANs) are other important examples of such networks. Unstructured wireless networks were introduced in the context of US DARPA [1] and PRNET [2] projects in early 1970s. The unstructured nature of such networks combined with their self-organizing properties originally made them attractive for defense and emergency response applications. Later on, unique characteristics of unstructured wireless networks expanded their applications into a wide area of other applications such as WSNs [3], VANETs [4,5], and pervasive computing networks [6]. While there are many definitions for unstructured ad hoc networks, they are all universally considered to have the following characteristics:

- There is no preexisting or fixed infrastructure;
- Such networks are dynamic due to mobility and allowing nodes to join or leave;
- Nodes are characterized by limited availability of resources;

*. *Corresponding author. Tel.: +1(949) 824-0380*
E-mail addresses: gharib@ipm.ir (M. Gharib); hyousefi@uci.edu (H. Yousefi'zadeh); movaghar@sharif.edu (A. Movaghar)

- Such networks offer poor physical security;
- Such networks have shared physical transmission media;
- Nodes are typically symmetric in the mentioned characteristics.

Fast growth of unstructured wireless networks combined with the entrance into sensitive and vital applications such as healthcare, emergency response, and military applications makes security requirements much more significant in such networks. Just like other networks, unstructured wireless networks need five basic security services including confidentiality, integrity, authentication, availability, and non-repudiation. Cryptography can offer all such services except availability. Availability may be achieved using other techniques such as Intrusion Detection Systems (IDS). Due to the absence of fixed infrastructure, the use of traditional key management systems such as Public Key Infrastructure (PKI) is very challenging and, at times, overhead prohibitive in unstructured wireless networks. The limitation of resources such as storage, process, and power in such networks further requires the use of a very efficient key management system. Hence, key pre-distribution systems appear to be more efficient and also more practical for unstructured wireless networks than other key management systems. As the cornerstone of cryptosystems, key pre-distribution requires the use of an effective two-layer routing algorithm in its practical implementations.

In the absence of infrastructured key management systems, the basic solution is to pre-load all network nodes with the whole set of keys. In this case, the key pre-distribution algorithm is called naive key pre-distribution. It is worth noting that either pairwise keys can be used for symmetric cryptosystems or public keys can be used for asymmetric cryptosystems. For symmetric cryptosystems, each node stores one pairwise key for communicating with each node using naive key pre-distribution. In asymmetric cryptosystems, each node stores the public key of all other nodes in order to be able to communicate with every node directly and securely. In such cases and when communicating with a destination node, a source node first encrypts its message with the key of the destination and then sends the message across the physical path. Since the message is encrypted, only the destination node can read the message.

Since it is not efficient to store all network keys in each node, it is preferred to store just k keys in each node with $k << n$ and n representing the number of network nodes. In case of naive key pre-distribution, the distributed keys form a fully connected graph in which each node has a direct link to all others. Thus, the source node can send a secure message to any other node directly and securely. In other cases where the number of stored keys is lower than the number of network nodes, there may not exist a direct link between the source node and the destination node. It means that the public key of the destination node or a pairwise key between the source and the destination node, may not be stored by the source node. Hence, pre-distributed keys form an overlay graph $G(E, V)$ in which V represents the set of network nodes, while E represents the set of secure links between nodes. Each link $e_{(i,j)} \in E$ represents a stored pairwise key between nodes i and j in the case of symmetric cryptosystems or a public key of node j stored in node i in the case of asymmetric cryptosystems. Clearly, using key pre-distribution schemes in unstructured wireless networks requires a more sophisticated routing algorithm applied to the overlay graph.

Key pre-distribution algorithms are categorized either as symmetric versus asymmetric cryptosystems or as deterministic versus probabilistic schemes. A symmetric key pre-distribution system is called deterministic if there exists at least one shared key between every pair of nodes. Otherwise, it is called probabilistic. Key pre-distribution systems are also categorized as random, polynomial, and combinatorial schemes. In this paper, we categorize them from the view point of overlay routing. Accordingly, key pre-distribution schemes fall into random graph, regular graph, or combinatorial graph categories. We review the history and the state of the art of each category and then compare different algorithms from the perspectives of performance and security strength.

The rest of the paper is organized as follows. Preliminaries including a brief description of general routing problem using key pre-distribution schemes along with important parameters of performance and security strength evaluation of key pre-distribution schemes are presented in Section 2. In Section 3.1, we review the literature of random-graph key pre-distribution schemes. Regular-graph key pre-distribution schemes are reviewed in Section 3.2. Section 3.3 contains a review of combinatorial-graph key pre-distribution schemes. A comparison of different key pre-distribution schemes is offered in Section 3.4. Section 4 provides a discussion of deterministic and probabilistic overlay routing algorithms. Finally, the paper is concluded in Section 5.

2. Preliminaries

In this section, we first define the general problem of routing for networks that rely on key pre-distribution schemes. Then, we define the most important parameters used in evaluating performance and security strength of key pre-distribution schemes. The notations used in this paper are defined in Nomenclature section.

In case of a symmetric cryptosystem and a key pre-distribution scheme with $k < n$, the overlay graph $G(V, E)$ is formed including n vertices representing network nodes, i.e. $|V| = n$. In this graph, there is a bidirectional link between each pair of nodes that share at least one common key. In such case, each pair of neighboring nodes can communicate directly and securely. Otherwise, a source node has to find a secure path in order to communicate with a destination. For this reason, the source node first uses a standard routing algorithm to find a physical path to the destination. Then, each node in the physical path checks whether it has an overlay link to its next neighbor on the physical path. If not, the node finds an overlay path to its neighbor. This operation requires sending the list of stored key IDs to physical neighbors. All nodes on this so called key-path have to be physically neighboring nodes. The operation of finding a secure path from the source node to the destination node, i.e. finding a physical path and then the key-path corresponding to each physical hop, is called key-path establishment. After completing the step of key-path establishment, the source node has a secure path to the destination node. Thus, the source node encrypts its message with the pairwise key agreed with its neighbor. In turn, the neighbor decrypts the message, encrypts it with the pairwise key of next neighbor, and sends the message. Each node does the same until the message reaches the destination node.

Figure 1 shows an instance of symmetric based key pre-distribution routing. In this example, each node stores three keys from a key pool containing 9 keys, i.e. $k_1, k_2, \ldots, k_9$. The source node is node number 1 and the destination node is node number 8. The blue links show the physical path identified using a typical routing algorithm. It reaches the destination node passing through nodes 3, 5, and 7. The red lines show the key-path, i.e., the overlay secure path. Since the source node and node 3 have a shared key, the key-path corresponding to the first physical path has just one hop. Since nodes 3 and 5 do not have a shared key, node 3 finds a corresponding key-path. Such key-path passes through nodes 2 and 4 reaching node number 5. Just like the corresponding key-path between nodes 3 and 5, the key-path between nodes 5 and 7 passes through nodes 4 and 6. Since node 7 and the destination node have a shared key, the key-path corresponding to the last physical hop includes just one overlay link. Considering the path from the source to the destination passes through node 4 twice, it can be argued that such path is not optimal. In this example, nodes 3, 2, 4, 5, 6, and 7 decrypt and encrypt the message, and thus they can read the message. Furthermore, the secure path is much longer than the physical path in the absence of key pre-distribution scheme. The length of the typical unsecured physical path is shown using parameter $\wp$.

In the case of asymmetric cryptosystems, each node stores public keys of k other nodes. Thus, the overlay graph $G(V, E)$ consists of n vertices and some connected links. The edge $e_{(i,j)} \in E$ represents the stored public key of node j in node i. Since storing of the public key of node j by node i does not guarantee that node j stores the public key of node i, the links in overlay graph are directed.

Using an asymmetric key pre-distribution scheme

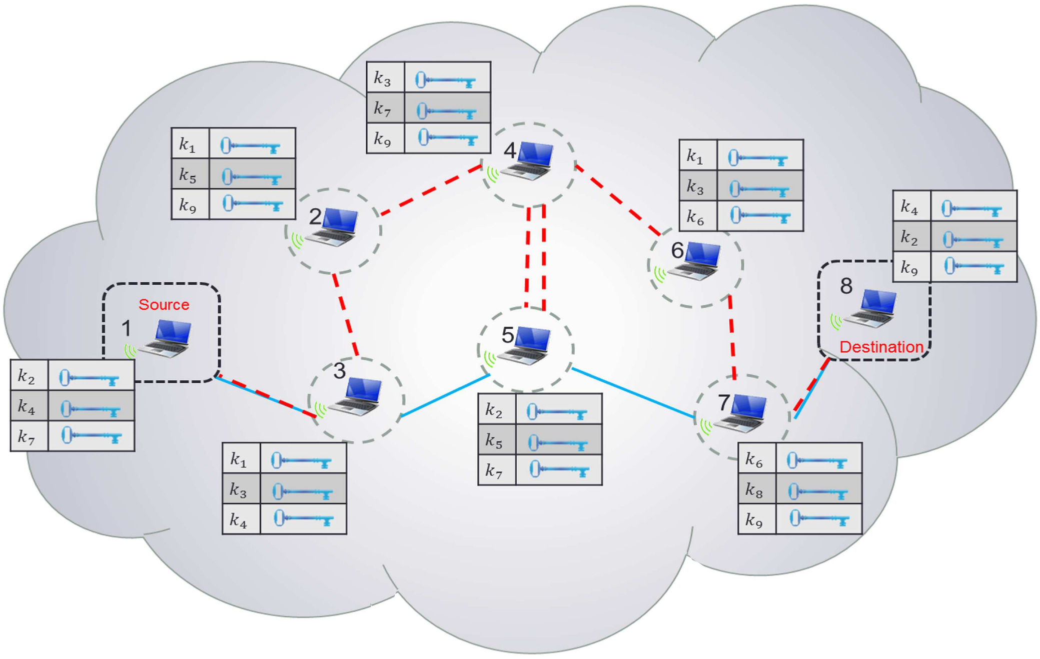


Figure 1. An example of overlay routing using symmetric key pre-distribution.

requires a different routing algorithm. In this case, the source node has to find an overlay path to the destination node. As the next step, the source node encrypts the message with the public key of the first overlay neighbor node and sends the message to it over the physical path. The physical path can be identified using any standard routing algorithm. The neighbor, in turn, decrypts the message using its private key and encrypts it again with the next overlay neighbor's public key. This operation is repeated until the message reaches the destination. It is worth noting that unlike symmetric schemes, in which all nodes participating in routing are able to read the messages, here, just the nodes on the overlay path can read the message. Another point of advantage of asymmetric key pre-distribution is related to routing in mobile networks in which overlay paths can stay intact while physical paths corresponding to each overlay hop may have to be changed.

Figure 2 represents an example of asymmetric key pre-distribution overlay routing. In this example, blue links represent the physical path while directed red links represent the overlay path. Each node stores just two public keys. As mentioned earlier, the source node first finds the key-path to the destination in the case of asymmetric key pre-distribution. In this example, there are two key-paths srepresented with red arrows. Using different routing algorithms, each one of the represented key-paths could be chosen. As the next step, each node inside the key-path finds the corresponding physical path toward the next node. In the case of choosing the key-path 1 → 2 → 3 → 6, each overlay hop includes just one physical hop. In the other case, i.e. key-path 1 → 5 → 6, the first key-path hop includes four physical hops, i.e. 1 → 2 → 3 → 4 → 5, while the second hop includes just one physical hop. It is clear that choosing the first key-path decreases the physical path length while choosing the other decreases the number of decryption-encryption steps.

In both symmetric and asymmetric systems, each intermediate decryption-encryption step increases the security risk and the probability of an adversary node capturing or changing a message. Hence, the *number of intermediate decryption-encryption steps* is considered as a security evaluation parameter. Another security evaluation parameter is the number of compromised nodes leading to compromising the whole network. This parameter is called *resiliency to node capture*. It is also observed that using a key pre-distribution scheme with $k < n$ may lead to loss of connectivity in an overlay graph. Hence, the *probability of network connectivity* is another important parameter in this context. It shows the average probability of existence of a path from a source node to a destination node. *Average key-path length* is yet another important performance evaluation parameter that is directly related to the choice of key pre-distribution scheme.

It is observed that a network could be considered as a two-layer graph under the paradigm of key pre-distribution schemes. While the bottom layer is formed by the physical routing layer, the top layer is a secure overlay layer formed by key distribution. Hence, the distribution of keys directly affects the performance and security of the network.

3. A categorical survey of key pre-distribution algorithms

3.1. Random graph key pre-distribution

Eschenauer and Gligor propose the original idea of

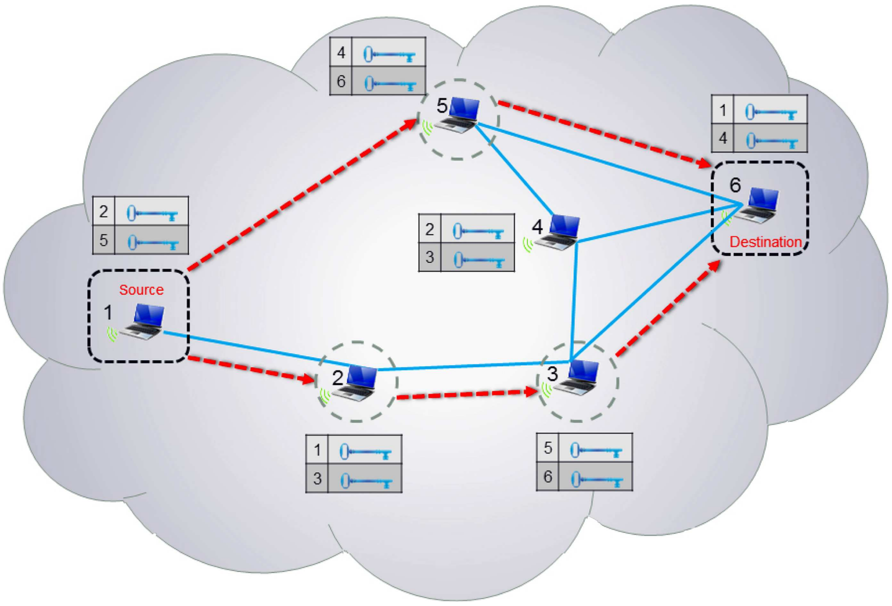


Figure 2. An example of overlay routing using asymmetric key pre-distribution [32].

random key pre-distribution for unstructured wireless networks [7]. The main idea is to pre-load each node with k randomly chosen pairwise keys. Pre-loaded keys are chosen uniformly from a key pool containing P keys. A lower value of P leads to a higher probability of two nodes sharing at least one key. Using such key pre-distribution scheme, each pair of neighboring nodes that share at least one key can communicate with each other securely. Clearly, the overlay graph forms a random graph in this scheme. Every node discovers its overlay neighbors by broadcasting an identifier list of its stored keys. Then, each node knows its overlay neighbors and also identifies the physical neighbors that do not store its shared key. In the next phase, referred to as path key establishment phase, each node finds an overlay path to those neighbors with which it does not have shared keys. The average probability that a shared key exists between two nodes, using the raw idea of random key pre-distribution, is called π and is equal to:

$$\frac{Ln(n)}{n}+\frac{c}{n}.$$

In this equation, n is the number of network nodes while c is a real constant. As mentioned earlier, this value is considered to be a very important performance parameter, since it directly affects routing performance. In essence, a higher value of π leads to a shorter overlay path. In finding a secure path to the destination, the source node first uses a standard routing algorithm such as AODV and then attempts at finding a corresponding key-path for each physical hop. Trivially, such two-layer routing approach can generate significant routing overhead traffic without guaranteeing to identify optimal routing path.

The basic idea of random key pre-distribution has attracted the attention of researchers due to its scalability, simplicity, flexibility, and usability. Accordingly, many key pre-distribution algorithms are proposed to extend the basic idea proposed by Eschenauer and Gligor enhancing its performance and security strength. Chan and Perrig [8] propose a q-composite random key pre-distribution. They suggest enhancing the security strength of the basic idea by requiring a secure connection to be held between those nodes that have at least q common shared keys. The pairwise key for secure communication, in this case, is a hash of shared keys. Clearly, such idea enhances security strength but dramatically decreases the probability of the existence of an overlay link between two nodes. Hence, the overlay graph is a random graph with a much lower number of links leading to much longer path lengths. Blom [9] proposes a symmetric key generation system in which each node needs to store just $(\theta + 1)$ keys where $\theta << n$ in order to generate a pairwise key with all other nodes directly and securely in a non-interactive manner. Liu and Ning [10] propose to implement the basic idea of [9] and generalize the algorithm to be used as a key pre-distribution scheme. They propose to generate a t-degree bivariate polynomial $f(x, y)$ with the property $f(x, y) = f(y, x)$. The polynomial is shown in Eq. (1) where the coefficients a_{ij}'s are chosen randomly and uniformly from the finite field F_q, where q is a prime number large enough to accommodate a cryptographic key.

$$f(x,y)=\sum_{i,j=0}^{t} a_{ij}x^i y^j. \tag{1}$$

Liu and Ning [10] propose to pre-load each node i with its share of polynomial calculated as $f(ID_i, y)$ in the network initialization phase. In this case, node i and node j can communicate with each other securely using their share of polynomial calculated based on the ID of the other node, i.e. the share of node i is equal to $f(ID_i, y)$ while the share of node j is $f(ID_j, y)$. Hence, the pairwise key between i and j is $f(ID_i, ID_j) = f(ID_j, ID_i)$. Using such idea, each node needs to store just one t-degree bivariate polynomial. Since the polynomial is unique, compromising $t + 1$ nodes lead to compromising the whole network. On the other hand, selecting a large value for t translates to a higher computational overhead. Thus, the value of t is an important tuning parameter addressing the tradeoff between performance and security. As an extension of their work and in order to improve security of their algorithm, Liu and Ning [10] also propose to form a pool with P bivariate polynomials of order t. Each node is pre-loaded with k randomly chosen polynomials from the pool. The polynomials are chosen similar to what is proposed in [7]. Each pair of neighboring nodes storing a common polynomial are able to communicate with each other directly and securely. It is worth noting that each polynomial requires $(t + 1)\log(q)$ storage space.

Du et al. [11] propose a key pre-distribution algorithm using multiple key spaces instead of a single key space proposed in Blom's symmetric key generation system [9]. In their extended work of [12], the same authors propose a key pre-distribution algorithm that requires a lower amount of memory but does not guarantee the formation of direct links in the overlay layer. The resiliency to node capture in the method of [11,12] is improved in comparison with the original idea of [9]. The authors further show that the resiliency of capture in their method is better than those of [7,8] for the same amount of storage. Gu et al. [13] propose a random key pre-distribution scheme based on the work of [7]. The authors propose to pre-load different nodes with a different number of keys, i.e. the number of keys is different among different key rings. Hence, the overlay graph $G(V,E)$ has a higher number of links in a number of nodes referred to as high-resilience nodes. They also prove that assuming the probability of node capture is the same for all nodes, an attack impact remains the same in both basic random key pre-distribution scheme of [7] and their heterogeneous scheme of [13]. The authors further propose to use high-resilience nodes as preferred intermediate nodes in the context of routing in order to reduce the overlay path length. It is worth noting that using such routing algorithms in unstructured wireless networks, especially in WSNs, results in faster consumption of energy in high-resilience nodes and potentially leads to loss of network connectivity.

3.2. Regular graph key pre-distribution

Several key pre-distribution schemes aim at improving performance and security strength of random distribution strategies by using an overlay graph that forms a regular graph. A regular graph [14] is defined as a graph in which all vertices have the same number of connected edges. Benefiting from the specific characteristics of a regular graph, different routing strategies are then proposed. Liu and Ning [10] propose to form an overlay in the shape of an $m \times m$ grid network where $m = \sqrt{n}$ in which each node is assigned to a specific intersection of the grid. Figure 3 shows the arrangement of nine nodes in an $m \times m$ grid according to the method of Liu and Ning [10]. Using the algorithm of [10], a pool is filled with $2m$ bivariate polynomials categorized into two separate groups called ${f_i}^c(x,y)$ and ${f_i}^r(x,y)$ for columns and rows, respectively, where $i = 0, 1, \ldots, m-1$. Each node is pre-loaded with two polynomials according to its position in the grid. As an instance, the node located at the third row and the fifth column of the grid is pre-loaded with ${f_3}^r(x,y)$ and ${f_5}^c(x,y)$. Interestingly, it is proven that the proposed method of [10] is equivalent to that of [11] utilized by the method of [12].

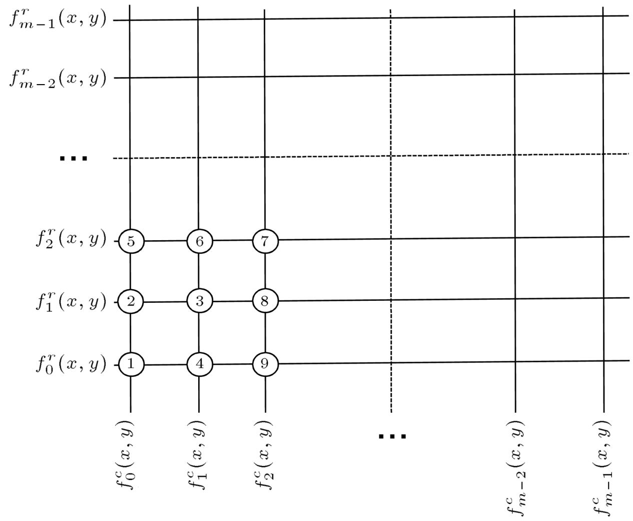


Figure 3. An example of the grid scheme of [10].

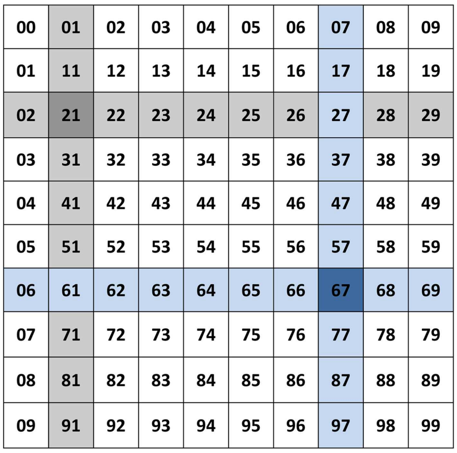
Using the key pre-distribution scheme proposed in [12], each node has to follow a routing algorithm in the overlay layer in order to communicate with other nodes. The source node first checks whether the destination node is in the same row or the same column as its own. If so, they can directly communicate. Otherwise, the source node has to encrypt the message with the pairwise key of a node located at its own row and the same column of the destination, or vice versa. An intermediate node decrypts the message, encrypts it again with the pairwise key of its own and the destination node. Clearly, using such algorithm requires at most one intermediate decryption-encryption step. It is worth noting that the two-dimensional grid-based algorithm described here could also be implemented as a higher-dimensional algorithm using a higher dimensional grid. In a higher-dimensional algorithm, the number of stored polynomials are exactly equal to the number of dimensions.

Later, Liu ei al. [15] proposed a d-dimensional polynomial-based key pre-distribution algorithm referred to as hypercube-based key pre-distribution. They propose to assign each node to a specific coordinate of the hypercube graph. As a result, each node is assigned a d-tuple ID, i.e. $(j_1, \cdots, j_d)$. Denoting the number of hypercube dimensions as d, the pool contains $d \times m^{d-1}$ randomly generated bivariate polynomials where:

$$m = \sqrt[d]{n}.$$

The assignment of polynomials to the nodes follows an algorithm in which each adjacent pair of nodes in the hypercube have exactly one common polynomial. As such, the overlay network forms a d-dimension hypercube. To find a path from the source to the destination node, the source node knows the key path toward the destination according to its position. The length of such path is exactly equal to the Hamming distance between the source and the destination node.

Chan and Perrig [16] propose Peer Intermediaries for Key Establishment (PIKE) algorithm in which each node stores $2 \times (\sqrt{(n)} - 1)$ keys. The overlay of this algorithm forms an $\sqrt{(n)} \times \sqrt{(n)}$ mesh network. Each node at this overlay shares a pairwise key with any node that lies in its row or in its column. Hence, each node can reach other nodes directly or through at most one intermediate node. Figure 4 shows an example of PIKE overlay graph for a network with 100 nodes. As an instance, in this figure, node 21 shares a pairwise key with each node located at the third row as well as each node located at the second column. PIKE overlay appears to be equivalent to 2-dimensional scheme of [10]. The authors of PIKE [16]



00	01	02	03	04	05	06	07	08	09
01	11	12	13	14	15	16	17	18	19
02	21	22	23	24	25	26	27	28	29
03	31	32	33	34	35	36	37	38	39
04	41	42	43	44	45	46	47	48	49
05	51	52	53	54	55	56	57	58	59
06	61	62	63	64	65	66	67	68	69
07	71	72	73	74	75	76	77	78	79
08	81	82	83	84	85	86	87	88	89
09	91	92	93	94	95	96	97	98	99

Figure 4. An example of PIKE scheme for a network with 100 nodes.

further extend their basic idea to form a higher d-dimensional mesh network in which each node has to store $d \times (\sqrt[d]{(n)} - 1)$ keys. In this case, each pair of nodes located at the same axis store a common pairwise key. Increasing number of the dimensions decreases the memory required for storing keys but increases the number of intermediate decryption-encryption steps.

Lee and Stinson [17] propose ID-based one-way function scheme (IOS). They suggest using an overlay graph referred to as a strongly regular graph with parameters $(n, \gamma, \Lambda, \mu)$. A strongly regular graph with the mentioned parameters is a loop-free regular graph with degree γ and n vertices. Further, each pair of adjacent vertices have exactly Λ common neighbors while all non-adjacent node pairs have exactly μ common neighbors. Hence, any pair of nodes have either a direct link or exactly μ two-hop key-paths.

Delgosha and Fekri [18-20] propose the multivariate polynomial-based key pre-distribution scheme (MKPS) as an extension of [15]. The basic idea is to generate a virtual d-dimensional hypercube similar to what is proposed in [15]. Then, each node is positioned in the intersection of different dimensions. Each node is assigned with a d-tuple ID corresponding to the node position. On the other hand, a distinct d-variate polynomial from a set of randomly generated d-variate polynomials is assigned to every hyperplane perpendicular to one of the axis lines. Each node is pre-loaded with d polynomials corresponding to its position, i.e. the intersection of hyperplanes. Each node can evaluate each d-variate polynomial at $(d - 1)$ dimensions calculating d univariate polynomials. Hence, the storage memory required is exactly the same as that of [15]. In this case, each pair of adjacent overlay neighbors whose Hamming distance is equal to one share exactly $(d - 1)$ univariate polynomial. To communicate securely with each other, two neighboring overlay nodes calculate all $(d - 1)$ shared polynomials at each other's dimension and generate a pairwise key as a combination of the $(d - 1)$ generated values. For a Hamming distance larger than one, the source node has to find a key-path to reach the destination. It is important to note that if the overlay arrangement of nodes complies with the physical arrangement, then the longest key-path is equal to d. However, the key-path could be much longer since there is no such compliance. The authors of [20] calculate the optimal value of d. An adversary node needs to compromise $(d-1)$ polynomials in comparison with the previous work in which only one polynomial is needed to be compromised. The pool contains $d \times m$ d-variate polynomials with $m = \sqrt[d]{n}$ for polynomials of degree $t < m$.

Çamtepe et al. [21] use an expander graph as the overlay graph in their key pre-distribution scheme. They propose to form Ramanujan expander graphs [22]. Ramanajun graphs are best known as asymptotically optimal explicit expander graphs providing the highest degree of expansion with the smallest degree of nodes. In the context of key pre-distribution, the use of such graphs results in achieving a higher degree of connectivity while storing a smaller number of keys. Figure 5 shows a Ramanujan expander graph $X^{\sigma,\upsilon} = X^{5,17}$ without showing self-loops and multi-edges. In this example, the graph has $\upsilon + 1 = 18$ vertices where each node is of degree $\sigma + 1 = 6$ including the self-loops and multi-edges. The authors of [21] propose to replace self-loops and multi-edges with randomly chosen edges such that each node stores the same number of keys.

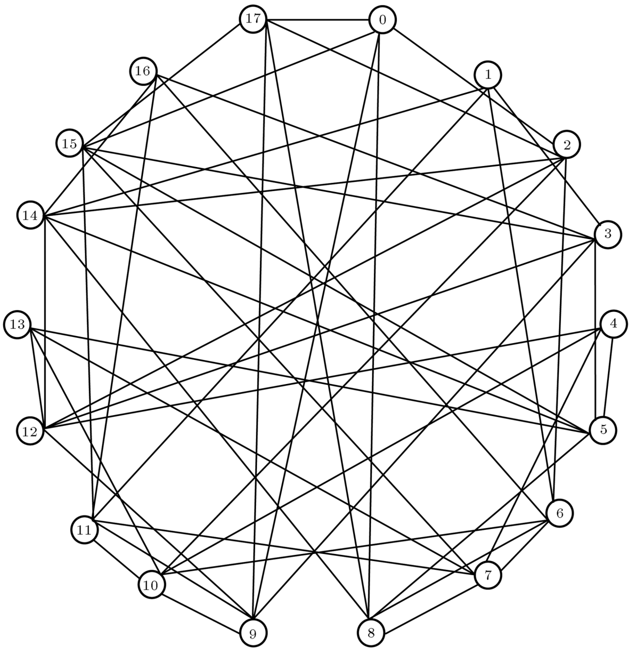


Figure 5. An example of Ramanujan expander graph $G(V, E) = X^{5,17}$ with 18 nodes where each node has 6 neighbors including self-loops and multi-edges. Self-loops and multi-edges are not shown in the figure.

Gharib et al. [23] propose Probabilistic Asymmetric Key Pre-distribution (PAKP) scheme. Built based on asymmetric key cryptosystems, this scheme pre-loads each node with k randomly chosen public keys of other nodes before the network deployment phase. The scheme is a random graph yet directed overlay key pre-distribution scheme. Hence, a k-regular directed graph $G(V, E)$ is formed with $|E| = k \times n$. In this graph, each node has k outgoing directed edges while the number of incoming edges is random. The authors show that in their scheme, the overlay graph is connected with a very high probability, even for small values of k. Further, the probability of connectivity is not significantly affected when increasing the number of network nodes. The authors also prove that the key-path length is in the order $O(\log_k n)$.

One of the major drawbacks of regular graph key pre-distribution schemes is that the assumption of maintaining a perfect regular graph during network lifetime may be violated as the result of random loss of some nodes. As an instance, recall the example of Figure 4. Consider nodes 61 and 27 as source and destination nodes, respectively. In such case, the connection between source and destination nodes passes through node 67 or node 21. A problem can occur when both of those nodes go down and, as the result, there is no overlay path between the source and the destination nodes.

3.3. *Combinatorial key pre-distribution*

Combinatorial design theory finds arrangements of subsets of a finite set such that certain characteristics are satisfied. Balanced Incomplete Block Design (BIBD) is a combinatorial design methodology used in key pre-distribution schemes due to its special characteristics [24]. BIBD arranges v distinct objects in b different blocks. Each object could be considered as a key inside the key pool while each block represents a key ring of a node. Each BIBD design is represented with a Boolean matrix named incidence matrix, containing v rows and b columns. A special case of BIBD design is represented by matrix (2). In this example, the key pool contains 9 keys equal to the number of incidence matrix rows. The network can have at most 12 nodes, because there are just 12 key rings associated with 12 columns. Since the first row of the incidence matrix contains four elements equal to one, the first key is shared among four nodes. Moreover, since the first column contains three elements equal to one, the first node stores three keys in its key ring. Having the incidence matrix, the overlay graph could be extracted easily considering each row of incidence matrix. In each row, there is a bidirectional link between each pair of nodes that have an element equal to one in that row. For instance, considering the first row of the incidence matrix example of matrix (2), there are bidirectional links between each pair of nodes 1, 6, 7, and 11 in the overlay graph. The overlay graph could also be shown using adjacency matrix. Considering a network with n nodes, the adjacency matrix is an $n \times n$ Boolean matrix in which each element, a_{ij}, i.e. the element at the i-th row and j-th column, represents whether there exists a directed link from node i to node j or not. Since overlay links in combinatorial key pre-distribution schemes are bidirectional, the adjacency matrix is always a symmetric matrix.

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0\\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0\\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1\\
0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1\\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0
\end{bmatrix} \tag{2}$$

Each BIBD design is expressed with a quintuplet (v, b, r, k, λ) where v and b are the number of objects (keys) and the number of blocks (key rings), respectively. Each object is repeated in exactly r distinct blocks and each block contains exactly k objects. It means that r nodes share a key and, further, there are exactly k keys in each key ring. Each pair of distinct objects occurs together in exactly λ blocks. Thus, each incidence matrix contains v rows and b columns, where each row contains exactly r one elements and, further, each column contains exactly k one elements. Any BIBD design can be expressed with the equivalent tuple (v, k, λ) because the relationship $bk = vr$ always holds.

Combinatorial design was first used as a key pre-distribution in unstructured wireless networks by Çamtepe and Yener [25] and then extended in [26]. The authors of [25,26] propose to use the symmetric design of BIBD to ensure full connectivity in overlay networks. Thus, their key pre-distribution scheme is a deterministic scheme. Symmetric BIBD design is a BIBD design in which $b = v$ and $r = k$. Defined as $(q^2 + q + 1, q + 1, 1)$, the proposed design is based on using a prime power parameter q. It contains $q^2 + q + 1$ keys and requires each node to store just $q + 1$ keys. Since $\lambda = 1$, this design guarantees to have exactly one common key for each pair of nodes. In such key pre-distribution method, an attacker can retrieve all keys by compromising just $q + 1$ nodes and knowing which node stores which key ring. An attacker can retrieve all keys by compromising at most $q^2 + 1$ nodes in the absence of such information. Moreover, such design is not scalable and as such cannot be used in large-scale networks.

A combinatorial trade or bitrade expressed by $\Upsilon(v, k)$ consists of sets $T = \{T_1, T_2\}$ where each T_i

contains η blocks of size k chosen from a finite set X such that the blocks of T_1 are completely disjoint from the blocks of T_2, i.e. $T_1 \bigcap T_2 = \phi$. Further, each set Υ chosen from X occurs in exactly the same number of blocks of T_1 as that of T_2. The volume of trade is equal to the number of blocks inside T_i where $|T_1| = |T_2|$. A trade is called Steiner if each set Υ chosen from X is repeated at most once in any of the sets T_1 and T_2. Furthermore, such Steiner trade is said to be strong if any block in T_1 and any block in T_2 intersect with each other in at most two elements. Ruj et al. [27] propose a method of constructing Strong Steiner Trades (SSTs) and prove that the proposed construction method results in a $2-(qk, k)$ SSTs with a volume of q^2 where q is a prime power number. The set of blocks $T_1 \bigcup T_2$ represents key rings, each containing k keys where $4 \leq k < q$ and the size of the key pool is qk. Such mapping from SST to key pre-distribution can generate $2q^2$ key rings. The authors suggest a proper value for k in the order $O(q) = O(\sqrt{n})$ where n is the number of nodes. According to the proposed algorithm of [27], two distinct neighboring nodes can communicate securely if each one of them is from a different set, T_1 or T_2, and if they store at least two common keys. The pairwise key between node A and node B is calculated as shown in Eq. (3), where k_1 and k_2 are the common keys. While SST establishes unique secret pairwise keys between nodes, the authors of [28] prove that the probability of sharing such a pairwise key does not exceed 0.25.

$$K_{AB} = K_{BA} = \text{hash}((k_1 \oplus k_2)\|id_A\|id_B). \qquad (3)$$

Unital design is a special asymmetric case of BIBD design. It is based on the value of variable δ for which the design is represented as $(\delta^3+1, \delta+1, 1)$ and contains $\delta^2(\delta^2 - \delta + 1)$ blocks. In the unital design, each block contains $\delta + 1$ objects and each object is repeated in $r = \delta^2$ distinct blocks. Since $\lambda = 1$, each pair of blocks have at most one key in common. Matrix (2) discussed earlier is, in fact, a special case of unital design for $\delta = 2$ and a representation of $(9, 3, 1)$.

Bechkit et al. [29] propose a key pre-distribution method based on unital design, to which they refer as Naive Unital Key Pre-distribution (NU-KP). The proposed scheme is extended and analyzed in [28]. NU-KP has a low key sharing probability in the order of $O(\frac{1}{k})$. In order to improve this probability, authors suggest pre-loading each node with ι completely disjoint blocks instead of just one block. Thus, the pairwise key between each pair of nodes is the hash value of the concatenated common keys. Referred to as ι-UKP method, the total number of nodes decrease to at most $\frac{b}{2}$ with each distinct pair shared between zero to ι^2 common keys. Increasing the value of ι in ι-UKP method leads to increasing the probability of sharing pairwise keys between nodes in the network, but decreasing the security strength of the network because each node receives more keys. Considering the fixed size of the key pool, storing more keys in each node allows an attacker to compromise a smaller number of nodes in order to retrieve all keys. For example, in the unital of matrix (2) and for 2-UKP, each node stores 6 keys out of 9. It means that an attacker needs to compromise just two nodes in order to retrieve all keys. There is also a practical disadvantage in implementing such method due to the difficulty of designing unitals for large values of δ.

3.4. A tabular comparison of key pre-distribution schemes

This subsection makes a categorical comparison among different key pre-distribution schemes. Table 1 provides a general comparison among different key pre-distribution schemes. Splitting different schemes in 3 overlay graph categories, the table covers the type of overlay graph, storage requirement, connectivity probability, node capture resiliency, and scalability of each scheme. Additional parameters of importance not included in the paper are discussed below. The first such parameter is the mobility support. We note that schemes built based on symmetric cryptosystems are not suitable for highly mobile environments. This is because high mobility breaks both physical and overlay paths in such networks. Hence, such systems can only perform well in networks with limited mobility. On the contrary, asymmetric-based key pre-distribution schemes are more suitable for use in mobile environments as mobility only affects connectivity of physical paths but not that of overlay paths.

As mentioned earlier, the key-path length is another parameter directly affecting both performance and security strength of a network. Considering the security aspect, each key-path hop needs an intermediate decryption-encryption step. The number of intermediate decryption-encryption steps in symmetric-based key pre-distribution schemes is in the order of physical path length, i.e. $O(\wp)$, while it is in the order of $O(\log_k n)$ in asymmetric-based schemes [23].

4. A comparison of overlay routing algorithms using key pre-distribution

Overlay routing algorithms are categorized under deterministic and probabilistic schemes. In deterministic schemes, every node has a direct overlay link to all other nodes. In such schemes, a source node just needs to find the physical routing path to a destination. In [30], Choi proposes a deterministic method using a hash function for key establishment pre-loading each node with just $\frac{n+1}{2}$ keys while ensuring all neighboring nodes have common keys. In deterministic key pre-

Table 1. A comparison of key pre-distribution schemes.

Category	**Scheme**	**Overlay graph**	**Storage**	**Connectivity probability π**	**Node capture resiliency**	**Scalability**
Random	Eschenauer and Gligor [7]	Random	$O(k)$	$\frac{Ln(n)}{n}+\frac{c}{n}$	$O(\frac{n}{k})$	Restricted by comm. overhead
	Q-composite [8]	Random	$O(k)$	$O(\log(\frac{Ln(n)}{n}+\frac{c}{n}))$	$O(\frac{n}{k})$	Restricted by comm. overhead
	Liu and Ning [10] (Storing k polynomials)	Random	$O(kt\log{(q)})$	$\frac{Ln(n)}{n}+\frac{c}{n}$	$O(\frac{n}{k})$	Restricted by comm. overhead
	Du et al. [12]	Random	$O(\theta\tau)$	$1-e^{-\frac{\tau^2}{\omega}}$	$O(\frac{n}{\theta\tau})$	Restricted by comm. overhead
	Gu et al. [13]	Random	Heterogeneous	$O(\frac{Ln(n)}{n}+\frac{c}{n})$	O (Number of high resilience nodes)	Restricted by comm. overhead
Regular	Liu and Ning [10]	$m\times m$ Grid	$O(t)$	$O(1/m)$	$O(m)$	$O(m^2)$
	Liu et al. [15]	Hypercube	$O(d)$	$O(d/\sqrt[d]{n})$	$O(d)$	$O(m^d)$
	PIKE [16]	$\sqrt{n}\times\sqrt{n}$ Mesh	$O(\sqrt{n})$	$O(1/\sqrt{n})$	$O(\sqrt{n})$	Better than random schemes
	IOS [17]	Strongly regular	$O(\gamma/2)$	$O(\gamma/n)$	$O(n/\gamma)$	$O(\gamma/2)$
	MKPS [20]	Hypercube	$O(d)$	$O(d/\sqrt[d]{n})$	$O(d)$	$O(m^d)$
	Çamtepe et al. [21]	Ramanujan expander	$O(\sigma)$	$\frac{\sigma+1}{\upsilon+1}$	$O(\upsilon/\sigma)$	$\upsilon+1$
	PAKP [23]	k-Regular	$O(k)$	≈ 1	n	Restricted by comm. overhead
Comninatorial	Çamtepe et al. [26]	Complete	$O(q)$	1	$q+1$	$O(q^2)$
	SST [27]	Non-regular	$O(k)$	$O(k^2/q^2)$	$O(q)$	$O(\eta^3)$
	NU-KP [28]	Non-regular	$O(\delta)$	$O(1/k)$	$O(\delta^2)$	$O(\delta^4)$
	ι-UKP [28]	Non-regular	$O(\iota\delta)$	$(1-e^{-1})<\pi$	$O(\delta^2)$	$O(\delta^4)$

distribution schemes, each node stores $O(n)$ keys. However, storing a large number of keys is not practical for nodes operating in unstructured wireless networks considering inherent storage limitations. Furthermore, deterministic schemes do not offer high resiliency against node capture. The latter is due to the fact that compromising just one node can lead to disclosing many keys and, consequently, compromising many links.

On the other hand, probabilistic key pre-distribution schemes require two layers of routing. Every two-layer routing algorithm has to consider performance metrics such as path length and security, i.e. the number of intermediate encryption-decryption steps. In key pre-distribution algorithms operating on random or combinatorial graphs, a source node has to find the physical path to its destination using a typical routing algorithm. As mentioned earlier, each node also finds the key-path to all other nodes. Such key-paths could be identified using any routing algorithm. As a result, a secure path from the source node to the destination node includes each physical hop's corresponding key-path. The main differentiating factor among key pre-distribution algorithms operating on a random graph is the probability of having a direct link between an arbitrary pair of nodes. In essence, a higher probability leads to having a shorter path. We remind the reader that using a two-layer routing algorithm may lead to a non-optimal key-path as noted in the example of Figure 1. It is also worth noting that using different routing algorithms for finding the physical path, and also the corresponding key-path, affects network performance.

As discussed in the Section 3.2, different regular graph-based key pre-distribution schemes require different routing algorithms to find the key-path corresponding to each key's physical hop. In [10], the XY routing algorithm [31] is used to find key-paths after forming an $m \times m$ grid as an overlay graph. The XY routing algorithm is used for two-dimensional mesh networks moving packets parallel to the X- and Y-axis until delivering them to the destination. The routing algorithm used for hypercube overlay network of [15] is based on the Hamming distance. This algorithm calculates the Hamming distance between the source and destination node IDs. Accordingly, at each step, a packet moves toward a dimension that shortens the Hamming distance. For instance, consider Figure 6, representing a 3-dimensional hypercube with 8 nodes. In this example, nodes $ID_1 = (1, 0, 0)$ and $ID_2 = (0, 0, 1)$ are source and destination nodes, respectively. The Hamming distance between the mentioned nodes is equal to two-hops because ID_1 and ID_2 differ in two dimensions. Hence, a packet moves from ID_1 in the direction of the first dimension reaching the node with the $ID = (1, 0, 1)$. In the next step, the packet moves in the direction of the third dimension to reach the destination. Since the algorithm of [16] uses an $\sqrt{n} \times \sqrt{n}$ mesh graph as the overlay, it uses the same XY routing algorithm to find the key-path corresponding to each physical hop. The algorithm of [20] also uses the same Hamming distance routing algorithm of [15]. The scheme of [17] forms an overlay using a strongly regular graph. As mentioned earlier, any pair of nodes that do not have a direct link have exactly μ common neighbors. Hence, each pair of physical neighbors have a direct key-path or at most a two hop key-path.

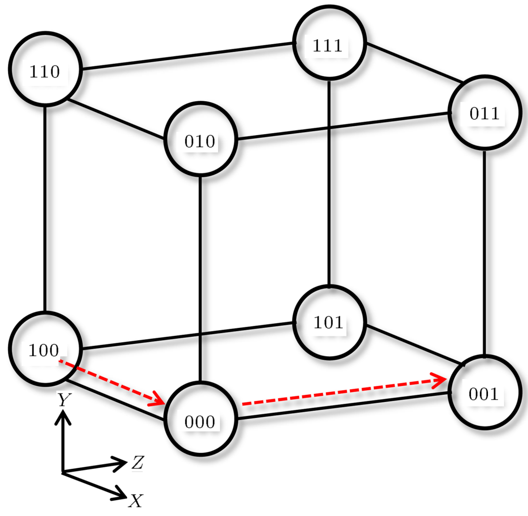


Figure 6. An example of hypercube routing.

Gharib et al. [32] propose an overlay routing algorithm for key pre-distribution schemes with the main advantage of being able to jointly optimize the costs of overlay and underlay paths. The other important advantage of this algorithm is being agnostic to the choice of key pre-distribution scheme. In that work, the authors model a network using a key pre-distribution scheme with a directed and weighted graph. In their model, all vertices and edges have their own costs. The weight of each edge represents the length of corresponding physical path in the case of asymmetric schemes and the length of corresponding key-path in the case of symmetric schemes. Further, a vertex weight represents the cost of decryption-encryption step. Considering the directed graph $G(V, E)$ with weighted edges and vertices, a Boolean linear optimization problem is proposed to find the lowest cost path considering both performance and security strength of the path. Each node is proposed to store a lookup table containing information about k stored keys where $k << n$. The lowest cost solution to the formulated problem is then identified in polynomial time. The limitation of the algorithm is expressed as the time complexity required for solving the relaxed LP problems in large-scale networks. The authors show that their algorithm works well for networks with the sizes of up to 500 nodes.

5. Conclusion

In this paper, we conducted a categorical review of key pre-distribution methods for unstructured wireless networks. Key pre-distribution schemes were categorized into random, regular-graph, and combinatorial schemes. We also compared different key pre-distribution schemes in terms of performance and security strength. We argued that random schemes were easier to implement in real world but required the use of efficient two-layer routing algorithms since they did not have any information about distributed keys. On the contrary, regular graph schemes only required an efficient underlay routing algorithm to operate but often utilized non-efficient routing paths. Last, we discussed combinatorial schemes which were easier to analyze because of their formal design but were subject to major implementation drawbacks in real-world networks.

We further provided a categorical overview of recent overlay routing algorithms relying on key pre-distribution. Overlay routing algorithms were classified under deterministic and probabilistic schemes. We noted that deterministic schemes only needed a single layer of routing since every node had a direct overlay link to all other nodes. The drawbacks of deterministic schemes were listed as the storage requirement associated with storing a large number of keys in each node and lack of resiliency to node captures. Consequently, probabilistic schemes were introduced as more practical alternatives of overlay routing capable of addressing both shortcomings of deterministic schemes at the cost of requiring two layers of routing. We then pointed to a recently introduced optimal overlay routing capable of scaling up to about 500 nodes. Improving the scalability of such scheme is the subject of active research.

Nomenclature

n	Number of network nodes
k	Size of a key chain
$G(V,E)$	Overlay graph
V	Set of vertices in $G(V,E)$ representing network nodes
E	Set of edges in graph $G(V,E)$ representing secure links
$e(i,j)$	Link between nodes i and j in graph G
P	Size of the key pool
π	Average probability of having an overlay link between two nodes
ID_i	ID of node i
$\wp$	Unsecured physical path length
q	Large prime number accommodating a cryptographic key
d	Dimension of multi-dimensional graphs
m	Parameter equal to $\sqrt[d]{n}$
t	Bivariate polynomial degree
ι	Number of blocks in UKP
δ	Order of a unital design
σ, υ	Parameters of Ramanujan expander graph
ω	Number of key spaces in multiple key space schemes
τ	Number of key spaces selected out of ω for use
η	Number of blocks of each disjoint set in [27]
$\theta + 1$	Number of per-node stored keys in [9,11,12]
γ	Degree of a strongly regular graph
Λ	Number of common neighbors of each pair of adjacent nodes in a strongly regular graph
μ	Number of common neighbors of each pair of non-adjacent nodes in a strongly regular graph

References

1. Toh, C.K., *Ad Hoc Wireless Networks: Protocols and Systems*, 1st Ed., Upper Saddle River, NJ, USA: Prentice Hall PTR (2001).

2. Haas, Z.J., Deng, J., Liang, B., Papadimitratos, P. and Sajama, S., *Wirelessad Hoc Networks*, 1st ed. New York, NY, USA: In Encyclopedia of Telecommunications, J. Proakis, Ed., John Wiley (2002).

3. Akyildiz, I., Su, W., Sankarasubramaniam, Y. and Cayirci, E. "A survey on sensor networks", *Communications Magazine, IEEE*, **40**(8), pp. 102-114, Aug. (2002).

4. Morris, R., Jannotti, J., Kaashoek, F., Li, J. and Decouto, D. "Carnet: A scalable ad hoc wireless network system", In *Proceedings of the 9th Workshop on ACM SIGOPS European Workshop: Beyond the PC: New Challenges for the Operating System*, ser. EW 9, New York, NY, USA: ACM, pp. 61-65 (2000).

5. Raya, M. and Hubaux, J.-P. "The security of vehicular ad hoc networks", In *Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks*, ser. SASN '05, New York, NY, USA: ACM, pp. 11-21 (2005).

6. Weiser, M. "The computer for the 21st century", *SIGMOBILE Mob. Comput. Commun. Rev.*, **3**(3), pp. 3-11, July (1999).

7. Eschenauer, L. and Gligor, V.D. "A key-management scheme for distributed sensor networks", In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS '02, New York, NY, USA: ACM, pp. 41-47 (2002).

8. Chan, H., Perrig, A. and Song, D. "Random key predistribution schemes for sensor networks", In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, ser. SP '03. Washington, DC, USA: IEEE Computer Society, p. 197 (2003).

9. Blom, R. "An optimal class of symmetric key generation systems", In *Proc. Of the EUROCRYPT 84 Workshop on Advances in Cryptology: Theory and Application of Cryptographic Techniques*, New York, NY, USA: Springer-Verlag, New York, Inc., pp. 335-338 (1985).

10. Liu, D. and Ning, P. "Establishing pairwise keys in distributed sensor networks", in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, ser. CCS '03. New York, NY, USA: ACM, pp. 52-61 (2003).

11. Du, W., Deng, J., Han, Y.S. and Varshney, P.K. "A pairwise key predistribution scheme for wireless sensor networks", In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, ser. CCS '03, New York, NY, USA: ACM, pp. 42-51 (2003).

12. Du, W., Deng, J., Han, Y.S., Varshney, P.K., Katz, J. and Khalili, A. "A pairwise key predistribution scheme for wireless sensor networks", *ACM Trans. Inf. Syst. Secur.*, **8**(2), pp. 228-258, May (2005).

13. Gu, W., Dutta, N., Chellappan, S. and Bai, X. "Providing end-to-end secure communications in wireless sensor networks", *Network and Service Management, IEEE Transactions on*, **8**(3), pp. 205-218, September (2011).

14. Chen, W.K., *Advanced Series in Electrical and Computer Engineering*, 5th Ed., 5 Toh Tuck Link, Singapore: World Scientific (1997).

15. Liu, D., Ning, P. and Li, R. "Establishing pairwise keys in distributed sensor networks", *ACM Trans. Inf. Syst. Secur.*, **8**(1), pp. 41-77, Feb. (2005).

16. Chan, H. and Perrig, A. "Pike: peer intermediaries for key establishment in sensor networks", In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings IEEE*, **1**, March, pp. 524-535 (2005).

17. Lee, J. and Stinson, D. "Deterministic key predistribution schemes for distributed sensor networks", In *Selected Areas in Cryptography, ser. Lecture Notes in Computer Science*, H. Handschuh and M. Hasan, Eds., Springer Berlin Heidelberg, **3357**, pp. 294-307 (2005).

18. Delgosha, F. and Fekri, F. "Key pre-distribution in wireless sensor networks using multivariate polynomials", In *Sensor and Ad Hoc Communications and Networks, 2005. IEEE SECON 2005. 2005 Second Annual IEEE Communications Society Conference on*, pp. 118-129, Sept. (2005).

19. Delgosha, F. and Fekri, F. "Threshold key-establishment in distributed sensor networks using a multivariate scheme", In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Proceedings*, pp. 1-12, April (2006).

20. Delgosha, F. and Fekri, F. "A multivariate key-establishment scheme for wireless sensor networks", *Wireless Communications, IEEE Transactions on*, **8**(4), pp. 1814-1824, April (2009).

21. Çamtepe, S., Yener, B. and Yüng, M. "Expander graph based key distribution mechanisms in wireless sensor networks", In *Communications, 2006. ICC '06. IEEE International Conference on*, **5**, pp. 2262-2267, June (2006).

22. Lubotzky, A., Phillips, R. and Sarnak, P. "Ramanujan graphs", *Combinatorica*, **8**(3), pp. 261-277 (1988).

23. Gharib, M., Emamjomeh-Zadeh, E., Norouzi-Fard, A. and Movaghar, A. "A novel probabilistic key management algorithm for large-scale manets", In *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*, pp. 349-356, March (2013).

24. Assmus, E.F. and Key, J.D., *Designs and Their Codes*, 1st Ed., New York, NY 10006, USA: Cambridge University Press (1992).

25. Çamtepe, S. and Yener, B. "Combinatorial design of key distributionmechanisms for wireless sensor networks", In *Computer Security ESORICS 2004, ser. Lecture Notes in Computer Science*, P. Samarati, P. Ryan, D. Gollmann, and R. Molva, Eds., Springer Berlin Heidelberg, **3193**, pp. 293-308 (2004).

26. Çamtepe, S. and Yener, B. "Combinatorial design of key distribution mechanisms for wireless sensor networks", *Networking, IEEE/ACM Transactions on*, **15**(2), pp. 346-358, April (2007).

27. Ruj, S., Nayak, A. and Stojmenovic, I. "Fully secure pairwise and triple key distribution in wireless sensor networks using combinatorial designs", In *INFOCOM, 2011 Proceedings IEEE*, pp. 326-330, April (2011).

28. Bechkit, W., Challal, Y., Bouabdallah, A. and Tarokh, V. "A highly scalable key pre-distribution scheme for wireless sensor networks", *Wireless Communications, IEEE Transactions on*, **12**(2), pp. 948-959, February (2013).

29. Bechkit, W., Challal, Y. and Bouabdallah, A. "A new scalable key pre-distribution scheme for wsn", in *Computer Communications and Networks (ICCCN), 2012 21st International Conference on*, pp. 1-7, July (2012)

30. Choi, T., Acharya, H.B. and Gouda, M. "The best keying protocol for sensor networks", In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on*, pp. 1-6 June (2011).

31. Zhang, W., Hou, L., Wang, J., Geng, S. and Wu, W. "Comparison research between xy and odd-even routing algorithm of a 2-dimension 3x3 mesh topology network-on-chip", In *Intelligent Systems, 2009. GCIS '09. WRI Global Congress on*, **3**, pp. 329-333, May (2009).

32. Gharib, M., Yousefi'zadeh, H. and Movaghar, A. "Secure overlay routing using key pre-distribution: A linear distance optimization approach", *Mobile Computing, IEEE Transactions on*, **15**(9), pp. 2333-2344 (2015).

Biographies

Mohammed Gharib is a Postdoctoral Researcher in the Department of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran. He received the BS degree from Baghdad University of Technology, Iraq, in 2007. He then received the MS and PhD degrees from Sharif University of Technology, Tehran, Iran, in 2009 and 2015, respectively. In September 2010, he joined Performance and Dependability Laboratory (PDL), where he worked toward his PhD degree, supervised by Professor Ali Movaghar, in Computer Engineering Department at Sharif University of Technology, Tehran, Iran. During the year 2014, he was a visiting research scholar in California Institute for Telecommunications and Information Technology, University of California Irvine, Irvine. His research interests include mobile ad hoc networks, wireless sensor networks, data networking, peer-to-peer networks, and their security aspects.

Homayoun Yousefi'zadeh received E.E.E and PhD degrees from the Department of EE-Systems at University of Southern California in 1995 and 1997, respectively. Currently, he is an Adjust Professor in the Department of EECS at University of California Irvine. In the recent past, he was a Consulting Chief Technologist at the Boeing Company and the founder as well as Chief Technology Officer of TierFleet. He is the inventor of several US patents, has published more than seventy scholarly reviewed articles, and authored more than twenty design articles associated with deployed industry products. Dr. Yousefi'zadeh is/was with the editorial board of IEEE Trans. Wireless Communications, IEEE Communications Letters, IEEE Wireless Communications Magazine, IEEE JSTSP, and Journal of Communications Networks. He was the founding Chairperson of systems' management workgroup of the Storage Networking Industry Association and a member of the scientific advisory board of Integrated Media Services Center at USC. He is a Senior Member of the IEEE and the recipient of multiple best paper, faculty, and engineering excellence awards.

Ali Movaghar is a Professor in the Computer Engineering Department at Sharif University of Technology, Tehran, Iran. He received his BS degree in Electrical Engineering from University of Tehran in 1977, and MS and PhD degrees in Computer, Information, and Control Engineering from the University of Michigan, Ann Arbor, in 1979 and 1985, respectively. He visited the Institute National de Recherche en Informatique et en Automatique in Paris, France, and the Department of Electrical Engineering and Computer Science at the University of California, Irvine, in 1984 and 2011; worked at AT&T Information Systems in Napervile, IL, in 1985-1986; and taught at the University of Michigan, Ann Arbor, in 1987-1989. His research interests include performance/dependability modeling and formal verification of wireless networks and distributed real-time systems. He is a Senior Member of the IEEE and the ACM.