Key Predistribution for Homogeneous Wireless Sensor Networks with Group Deployment of Nodes

Keith M. Martin,Maura B. Paterson*
{keith.martin, m.b.paterson}@rhul.ac.uk
Information Security Group
Department of Mathematics
Royal Holloway, University of London
Egham, Surrey, TW20 0EX, U.K.

Douglas R. Stinson†
dstinson@uwaterloo.ca
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

September 26, 2008

Abstract

Recent literature contains proposals for key predistribution schemes for sensor networks in which nodes are deployed in separate groups. In this paper we consider the implications of group deployment for the connectivity and resilience of a key predistribution scheme. After showing that there is a lack of flexibility in the parameters of a scheme due to Liu, Ning and Du, limiting its applicability in networks with small numbers of groups, we propose a more general scheme, based on the structure of a resolvable transversal design. We demonstrate that this scheme permits effective trade-offs between resilience, connectivity and storage requirements within a group-deployed environment as compared with other schemes in the literature, and show that group deployment can be used to increase network connectivity, without increasing storage requirements or sacrificing resilience.

Keywords: Group-based deployment, key predistribution, wireless sensor networks

1 Introduction

A sensor node is a battery operated device for measuring some physical quantity (such as temperature or humidity) that is capable of engaging in wireless communication. Such sensors are deployed to take measurements throughout a target area, whereupon they communicate with each other in order to transmit and process the collected data. The resulting ad hoc network is known as a *wireless sensor network* (WSN). Such networks can be employed for a wide range of applications [25], whether scientific, commercial, humanitarian or military. The data being transmitted over the wireless medium is frequently valuable or sensitive; hence, there is a need for cryptographic techniques to provide data integrity, confidentiality and authentication. However, sensor nodes are highly constrained in terms of their memory, computational abilities and battery power; as such, symmetric cryptographic primitives are preferred in many instances to public key techniques whose computational requirements are regarded as being an excessive burden for the sensors. This in turn necessitates the sharing of keys by the nodes.

Key predistribution is one widely-studied solution to the problem of key establishment in sensor networks: keys (or other keying material) are stored in the nodes' memories prior to deployment, so

*Research supported by EPSRC grant EP/D053285/1
†Research supported by NSERC grant 203114-06

that nodes that share keys can then communicate securely once deployed (provided they are within communication range). A *key predistribution scheme* (KPS) is a means of specifying which nodes store which keys. Many such schemes have been proposed for use in WSNs (see [4, 23, 27] for surveys of this field); they essentially involve a trade-off between the competing requirements of low memory usage, high network connectivity, and resilience against adversaries who capture nodes and extract the keys that they store. The majority of schemes in the literature are intended for application in networks in which the sensors' final location is unknown prior to deployment (as would be the case if the nodes were scattered from an aeroplane). We refer to this as the *standard model* of a WSN. However, the wide range of potential applications for sensor networks means that networks used in practice do not always conform to this particular model. Martin and Paterson [23] have classified sensor network environments according to the extent to which there is control over the sensors' locations; in this paper we are interested in networks that fall into their category of having fixed sensors with partial location control. Specifically, we consider networks in which sensors are deployed in groups such that sensors from a group are closer together on average than sensors from different groups. We refer to this as *group deployment* of sensors[1].

Group deployment may be used in order to improve the coverage of the target region by sensors, as it provides more control over the physical distribution of sensors. Alternatively, it may be a convenient way of carrying out the deployment: in the case where several vehicles are available for distributing sensors, they could be used to deliver sensors to different portions of the target area simultaneously. However, the main motivation for considering group deployment from the point of view of key distribution is the fact that the partial location knowledge it provides can be used in order to improve the connectivity of the network. In this paper we consider how this information can best be exploited to achieve a KPS with the properties we desire.

The use of group deployment in a network has implications for the design of a KPS. In order to analyse fully the properties of a network in which a KPS is deployed, it is necessary to consider both the *block graph*, whose vertices are nodes, with pairs of nodes that share keys being joined by edges, and the *physical graph*, whose vertices are again nodes, but in this case they are joined by edges whenever they are within communication range after deployment (to use the terminology of [16]). These intersect to yield the *key-sharing graph*, whose edges represent pairs of nodes that can communicate securely after deployment as they are both within range and share keys. In the group deployment scenario, not only is the physical distribution of the nodes slightly different from that in the standard model, but there is also a degree of information about the physical graph, as the groups to which each node belongs are known before they are deployed. This information can be used to adjust the block graph in order to improve the properties of the resulting key-sharing graph.

In designing a KPS we seek to achieve a network with good connectivity, but also good resilience: as the nodes operate unattended and lack tamperproof hardware, they are vulnerable to physical attack. Once an adversary extracts a key from a sensor node, any further communication links that use the same key are no longer secure. As in [16], we model the extent to which a KPS can withstand an adversary that compromises a number of nodes by the quantity $\mathsf{fail}(s)$, which denotes the probability that a secure link between two uncompromised nodes becomes insecure after s nodes have been captured uniformly at random. The lower the value of $\mathsf{fail}(s)$ for each s, the more resilient the KPS. Thus an effective KPS is one that achieves a high level of connectivity, with small $\mathsf{fail}(s)$.

1.1 Group Deployment in the Literature

Several authors have proposed key predistribution schemes for sensor network environments that could be treated as examples of group deployment. These schemes can be considered to belong to one of three broad categories: those involving homogeneous networks, those in which there is additional knowledge of the groups' locations, and those designed for heterogeneous networks. We briefly survey these categories for the sake of completeness, although the rest of the paper is exclusively concerned with the case of a homogeneous network in which there is no a priori knowledge of the groups' locations.

[1]Note that in this paper the term "group" relates simply to the physical distribution of the sensors; the shared keys we are considering are intended for pairwise communication between sensors.

1.1.1 Standard-Model Schemes

One possible approach to key predistribution for group-deployed networks is to employ a standard-model KPS; however, we would expect their performance to be exceeded by that of schemes designed to take group deployment into account (cf. Section 4.2). Nevertheless, such schemes can be used as components in the construction of group-based KPSs; here we briefly describe some relevant examples.

Single Key Perhaps the simplest example of a KPS is the scheme in which a single key is stored by each node in the network. This provides optimal connectivity and storage, but has poor resilience, as all communication links become insecure when a single node is captured.

Distinct Pairwise Keys In this scheme, a unique key is assigned to each pair of nodes. This scheme has optimal resilience, since compromise of a node does not affect the security of any keys shared by uncompromised nodes. However, in a network with n nodes, it requires each node to store $n-1$ keys, which is infeasible if n is large.

Eschenauer and Gligor's KPS [13] This scheme is a probabilistic KPS, with each node drawing m keys uniformly without replacement from some finite keypool $\mathcal{K}$. The value of m and size of $\mathcal{K}$ can be chosen in order to ensure any pair of nodes have a certain probability of sharing a key.

Blom's Scheme [1, 2] This scheme uses a symmetric bivariate polynomial over some finite field $\mathrm{GF}(q)$, i.e. a polynomial $P(x, y) \in \mathrm{GF}(q)[x, y]$ with the property that $P(i, j) = P(j, i)$ for all $i, j \in \mathrm{GF(q)}$. A node with ID i stores a *share* in P, consisting of the univariate polynomial $f_i(y) = P(i, y)$. In order to communicate with node j, it computes the common key $K_{ij} = f_i(j) = f_j(i)$; this process enables any two nodes to share a common key. If P has degree t, then each share consists of a degree t univariate polynomial; each node must then store the $t+1$ coefficients of this polynomial. These are elements of $\mathrm{GF}(q)$, as are the pairwise keys that are established; thus, storing a degree t share requires as much space as storing $t+1$ keys. If an adversary captures s nodes, where $s \leq t$, then it does not learn any information about keys established between uncompromised nodes; however, if it captures $t+1$ or more nodes then it can interpolate to compute the polynomial P and hence learn all the keys.

Lee and Stinson's Transversal Design KPS [16] Lee and Stinson propose a KPS that they refer to 7as a 'linear scheme' based on a combinatorial structure known as a *transversal design* (see Section 3.2 for a precise definition), whose parameters can be chosen to vary the resilience, connectivity, and storage requirements of the KPS. They also describe how the resilience of the scheme can be increased by combining a transversal design with Blom's scheme to construct a KPS, which they term a 'multiple space scheme'.

1.1.2 Homogeneous Schemes

Liu, Ning and Du explicity use the term "group-based key predistribution" [20, 21]. They model group deployment of nodes by supposing the nodes in a group follow a Gaussian distribution about some mean position, with the mean positions of the groups distributed uniformly at random throughout a target area.

In their KPS, all pairs of nodes within a group can establish a key. The different groups are connected by means of keys distributed within *cross groups*, sets of nodes such that each cross group contains one node from each group, and such that every node is contained in precisely one group; the key distribution is thus based on a grid structure. The examples they give suggest they have in mind networks in which the number of nodes in a group is approximately the square root of the total number of nodes. Keys are distributed within each group and each cross group by means of either a distinct pairwise KPS, or an instance of Blom's scheme.

Group deployment in a homogeneous environment was also considered by Zhou, Ni and Ravishankar [29]. In their scheme, nodes within each group are assigned distinct pairwise keys, and for every pair of groups there are t pairwise keys shared by one node from each of the groups.

1.1.3 Location-Aware Schemes

Group deployment has also appeared in the literature in the context of networks where the groups are deployed within regions consisting of regular polygons that tessellate the plane, and where the location of each group is known before deployment.

Schemes combining a square grid with polynomial-based key predistribution are discussed in [12, 14, 19, 22, 28]; a square grid with random key predistribution was considered in [5], and the combination of a square grid with a scheme due to Lee and Stinson was considered in [26]. Papers describing the use of hexagonal grids with polynomial-based key predistribution include [3, 9–11, 17, 18, 22, 31]; hexagonal grids combined with random key predistribution were analysed in [24]. Polynomial-based key predistribution in triangular grids was proposed in [7, 30].

1.1.4 Heterogeneous Schemes

Heterogeneous sensor networks in which each group contains at least one more capable node have also received some attention. Typically, the more powerful nodes are considered to act as cluster heads in the formation of a hierarchical network, with each lower level sensor node in a group communicating (either directly, or via other lower level nodes) with the relevant cluster head, which then communicates with a base station. Papers considering this model include [8, 15].

1.2 Our Contributions

In Section 2 of this paper we analyse what properties are necessary for a KPS to perform well in a homogeneous network with group deployment. Section 2.1 focuses on connectivity requirements; in particular, we consider the desirability of balancing the connectivity between groups with that occurring within the groups, and examine how to vary the key sharing probabilities of a KPS in order to achieve this. Section 2.2 is devoted to an analysis of the requirements posed by groups of various sizes.

In Section 3 we propose a new KPS designed to suit a group-deployed environment. After observing certain limitations in a scheme due to Liu, Ning and Du [20, 21], we construct a more flexible scheme, motivated by the considerations of Section 2, which addresses these issues.

We analyse the behaviour of this scheme in Section 4, illustrating how group deployment can lead to enhanced connectivity in Section 4.1, and demonstrating the improved performance of our KPS compared with others from the literature in Section 4.2. We conclude in Section 5 with a discussion of the attributes that make a KPS suitable for use with group deployment.

2 Group Deployment in Homogeneous WSNs

Of the possible WSN environments that involve group deployment, the one that is derived most immediately from the standard model is that in which the group to which each node belongs is known, but there is no knowledge of the groups' eventual locations prior to deployment. This might arise, for example, in a situation where nodes are distributed in batches from an aeroplane, or when more than one aircraft are used to deploy groups of nodes. The network formed after deployment is homogeneous, with all nodes having an equivalent role. In this respect it does not differ from the standard model; in particular, the requirements that the resulting network be connected and resilient to node capture are the same. However, this alternative model of physical deployment has different implications for the way a KPS must be constructed in order to achieve these requirements. In this section we consider some of these requirements, with a particular focus on connectivity, and we analyse how the desired patterns of connectivity can be achieved by adjusting the probabilities for nodes to share keys in a KPS.

2.1 Connectivity in a Homogeneous Environment

In a homogeneous network, we wish for any two nodes to be able to communicate with each other across the network (potentially using several intermediate hops). This is possible when the key-sharing graph is a connected graph; as in [16], we refer to this as the *global connectivity* of the network. (Alternatively, we may consider it satisfactory if a vast majority of the nodes lie in one connected component of the key-sharing graph.) In addition, we also wish to consider the *local connectivity*: the probability that neighbouring nodes can communicate securely and efficiently. The global connectivity of a network is greatly affected by the physical topology, and thus it can be difficult to assess based on the properties of KPS. Local connectivity is a useful measure as it permits us to abstract away some of this dependence on the nodes' distribution, yet it still gives us a useful measure of the performance of a KPS, because a scheme with a high local connectivity is more likely to lead to a globally connected network. Two nodes that are within range can communicate securely if they share a common key. In addition, two neighbouring nodes that do not share a key may be able to communicate securely with the assistance of an intermediate node that is within range of, and shares a key with, both of the nodes. This is referred to as a *two-hop path*. In designing a KPS, we would like to ensure that neighbouring nodes have a high probability of being able to communicate securely either directly or via a two-hop path.

The above considerations apply equally in the standard model and the group deployment model. When using group deployment, however, there is the potential for the patterns of key sharing within a group, and between separate groups, to be quite different. Nevertheless, the groups arise simply as a feature of how the network is deployed; the network still forms a homogeneous whole, and we would like any two neighbouring nodes to be able to communicate securely with high probability, regardless of whether they come from the same group, or from different groups. In order to avoid communication bottlenecks, or the risk that an entire group could become disconnected from the rest of the network, we would like to ensure that the probability of nodes from different groups being able to communicate securely is similar to that of nodes from within a group. We will refer to this as *balanced local connectivity*. In Section 2.1.1, we examine how to achieve balanced local connectivity by altering the key sharing probabilities between nodes from the same group, as compared to nodes from different groups.

2.1.1 Balancing the Two-Hop Connectivity

Consider a KPS for a group deployed network for which two nodes from the same group have a probability p of sharing a key, whereas nodes from different groups have probability q of sharing a key. We would like to know how p and q must be related in order for the network to achieve balanced local connectivity.

The probability that two neighbouring nodes share a key directly is either p or q, depending on whether they are from the same group or from different groups. The probability that two neighbouring nodes can communicate via a two-hop path depends on the nodes' physical distribution as well as p and q, as it depends on the number of additional nodes that lie in range of both of them. In order to avoid having to make assumptions about the specific distribution of the nodes, we suppose that a node A in some group $\mathcal{G}$ and a node B that is a neighbour of A have, on average, η common neighbours in $\mathcal{G}$. In particular, this implies that two neighbouring nodes from the same group have η common neighbours, and two neighbouring nodes from different groups have η common neighbours from each of the two groups. The parameter η was used in the standard model in [16].

We will now determine the probability that two neighbouring nodes can communicate via a one-hop or two-hop path. For the sake of simplicity, we ignore two-hop paths involving nodes from three different groups, as these occur with low probability; similarly we ignore two-hop paths between two nodes within a group where the intermediate node belongs to another group.

Lemma 2.1. *If two nodes from one group are neighbours, then the probability that they can communicate securely via a one-hop or a two-hop path is* $1 - (1 - p)(1 - p^2)^\eta$.

Proof. Consider a pair of neighbouring nodes from the same group, as shown in the diagram on the left in Figure 1. The probability that they don't share a key is $1 - p$. For each common neighbour, the probability that a two-hop path cannot be established through that neighbour can be approximated by $1 - p^2$ (if we assume that the probabilities of any pair of nodes sharing a key are independent). Thus the probability that the nodes do not share a key and cannot communicate via a two-hop path is $(1 - p)(1 - p^2)^\eta$; the result then follows immediately. □

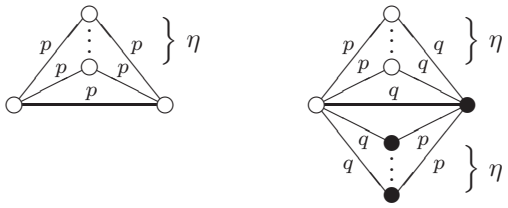


Figure 1: One-hop and two-hop paths between two nodes of the same group (left), and two nodes from different groups (right). White and black circles represent nodes from two different groups; edges are marked with the probability that the two corresponding nodes share a key.

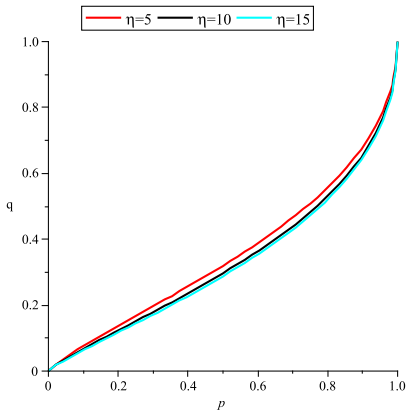


Figure 2: Values of p and q ensuring balanced local connectivity for $\eta = 5$, $\eta = 10$, $\eta = 15$

Lemma 2.2. *If two nodes from different groups are neighbours, then the probability that they can communicate securely via a one-hop or a two-hop path is* $1 - (1 - q)(1 - pq)^{2\eta}$.

Proof. The diagram on the right in Figure 1 shows two neighbouring nodes from different groups; the probability they do not share a key is $1 - q$. These nodes have η common neighbours from each of their groups. For each such neighbour, the probability the nodes cannot communicate via a two-hop path through that neighbour is $(1 - pq)$. Hence the total probability they do not share a key and they cannot communicate via a two-hop path is $(1 - q)(1 - pq)^{2\eta}$. Therefore the probability that they can communicate via a one-hop or two-hop path is $1 - (1 - q)(1 - pq)^{2\eta}$. □

From this we conclude that the probability that two nodes that are within range can communicate via a one-hop or two-hop path will be the same for all pairs of nodes that are within range precisely when

$$(1 - p)(1 - p^2)^{\eta} = (1 - q)(1 - pq)^{2\eta}. \tag{1}$$

Figure 2 illustrates this relationship between p and q for various values of η. We see that for balanced connectivity we require q to be slightly less than p. This is in contrast to key distribution in the standard model in which q necessarily equals p. Although nodes from different groups are less likely to be in communication range, when they do fall within range then they are likely to have more common neighbours that could be used to form a two hop path; hence, the desired probability of being able to communicate securely can be obtained using a lower key-sharing probability.

Taking logarithms of both side of Equation 1, we can express as a function of p and q the value of η that leads to balanced local connectivity:

$$\eta = \frac{\ln(1 - p) - \ln(1 - q)}{2\ln(1 - pq) - \ln(1 - p^2)}. \tag{2}$$

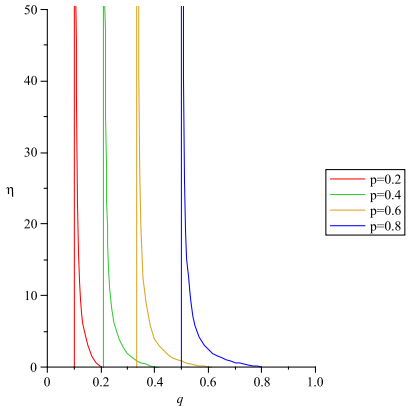


Figure 3: Values of η and q ensuring balanced local connectivity for $p = 0.2, 0.4, 0.6, 0.8$

Figure 3 shows the relationship between values of η and q that ensure Equation 2 is satisfied for various values of p.

For fixed p, Equation 2 expresses η as a function of q. This function has a vertical asymptote at $q = 1/p - \sqrt{1/p^2 - 1}$, is equal to zero when $q = p$, and is positive for all q between these bounds. Thus, for a given value of p, we know that balanced local connectivity can be achieved by taking some value of q between $1/p - \sqrt{1/p^2 - 1}$ and p. A value, or range of possible values, for η, determines the appropriate q.

Example 1. Consider a network in which the expected value of η is 7. Suppose we wish the probability for two neighbouring nodes to communicate securely using a one-hop or two-hop path to be 0.98. This can be achieved in a balanced manner by using a KPS with $p = 0.6$ and $q = 0.37$.

2.2 Number of Groups Deployed

In a network with λ groups and a nodes in each group, the proportion of pairs of nodes that belong to the same group is $(a-1)/(a\lambda-1)$, and the proportion of pairs of nodes from different groups is $(\lambda-1)a/(a\lambda-1)$. The relative proportions of such pairs of nodes depend on the values of a and λ; in turn this affects the number of in-group or cross-group keys that must be employed to achieve desired patterns of connectivity. In some scenarios, the number of groups to be deployed is determined by the application or by the available means of deployment, whereas in others it may be possible to choose the value of λ. In Section 4.1, we will see an example of how increasing the number of deployment groups can lead to an increase in network connectivity. However, increasing the number of groups is likely to increase the costs associated with deploying the nodes. In particular, the deployment of a large number of small groups is likely to be infeasible in most instances. Hence, the case where $\lambda \leq a$ is of the greatest practical interest, and so in Section 3 we restrict our attention to networks with n nodes deployed in λ groups, where $\lambda \leq \sqrt{n}$.

3 Construction of KPSs for Networks using Group Deployment

Having examined the properties required of KPSs in a group deployment context we now consider how to construct KPSs with these properties. We begin by examining a construction due to Liu, Ning and Du [20, 21], and observe that its restricted set of parameters limits the extent to which its properties can be adapted as desired. We then discuss a more flexible approach that permits trade-offs between resilience and connectivity in networks of n where the number λ of groups is at most $\sqrt{n}$.

3.1 Limitations of Liu, Ning and Du's Group-Based KPS

A KPS for networks using group deployment was proposed by Liu, Ning and Du in [20, 21]. In their scheme they partition the set of nodes into *cross groups*, with each node lying in one cross group, and

λ	in-group	cross-group
1	1	0.001
2	1	0.002
4	1	0.004
8	1	0.007
16	1	0.015
32	1	0.029
64	1	0.057
128	1	0.111

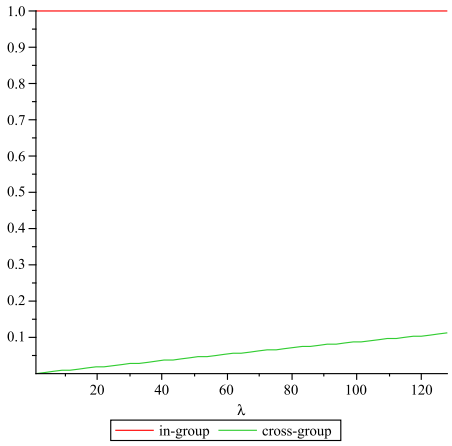


Figure 4: The probability that neighbouring nodes from within a group or from different groups can communicate securely using the group-based scheme of Liu, Ning and Du in a network of 16384 nodes deployed in λ groups, with $\eta = 7$.

each cross group containing precisely one node from each group. They propose distributing keys within each group and each cross group by one of two methods:

- assigning a distinct key to each pair of nodes,
- using Blom's scheme [1, 2].

Example 2. Consider a network of sixteen nodes, deployed in four groups of four nodes. We list an example of the keyrings of the nodes when distinct pairwise keys are used in the KPS of Liu et al.

group 1	group 2
$\{k_1, k_2, k_3, k_{25}, k_{26}, k_{27}\}$	$\{k_1, k_4, k_5, k_{31}, k_{32}, k_{33}\}$
$\{k_7, k_8, k_9, k_{25}, k_{28}, k_{29}\}$	$\{k_7, k_{10}, k_{11}, k_{31}, k_{34}, k_{35}\}$
$\{k_{13}, k_{14}, k_{15}, k_{26}, k_{28}, k_{30}\}$	$\{k_{13}, k_{16}, k_{17}, k_{32}, k_{34}, k_{36}\}$
$\{k_{19}, k_{20}, k_{21}, k_{27}, k_{29}, k_{30}\}$	$\{k_{19}, k_{22}, k_{23}, k_{33}, k_{35}, k_{36}\}$

group 3	group 4
$\{k_2, k_4, k_6, k_{37}, k_{38}, k_{39}\}$	$\{k_3, k_5, k_6, k_{43}, k_{44}, k_{45}\}$
$\{k_8, k_{10}, k_{12}, k_{37}, k_{40}, k_{41}\}$	$\{k_9, k_{11}, k_{12}, k_{43}, k_{46}, k_{47}\}$
$\{k_{14}, k_{16}, k_{18}, k_{38}, k_{40}, k_{42}\}$	$\{k_{15}, k_{17}, k_{18}, k_{44}, k_{46}, k_{48}\}$
$\{k_{20}, k_{22}, k_{24}, k_{39}, k_{41}, k_{42}\}$	$\{k_{21}, k_{23}, k_{24}, k_{45}, k_{47}, k_{48}\}$

There are 48 keys in total, each key is stored by two nodes, and each node stores six keys. Keys k_1, k_2 through to k_{24} are used for cross-group communication, keys $k_{25}, k_{26}, \ldots, k_{48}$ are in-group keys. Any pair of keyrings in the same row, or in the same column, intersect in precisely one key.

The distinct-pairwise scheme is optimal from the point of view of resilience, but has large storage requirements. If Blom's scheme is used with polynomials of degree t, then nodes must store the equivalent of $t + 1$ keys, but all keys within a particular group or cross group are secure unless $t + 1$ or more nodes from that group/cross group are captured. This permits a trade-off between the storage requirements and the resilience of the scheme.

The connectivity of the scheme, however, is entirely determined by the number and size of the groups. When n^2 nodes are deployed in λ groups, the total number of pairs of nodes that share keys is $\lambda\binom{n^2/\lambda}{2} + \frac{n^2}{\lambda}\binom{\lambda}{2}$, and nodes within a group share a key with probability 1, while nodes from different groups share a key with probability λ/n^2. In the case where the number of groups is small, this probability is very low. For example, Figure 4 shows the respective probabilities that in-group neighbors and cross-group neighbours can communicate securely using a one-hop or a two-hop path in a network of 16384

nodes deployed in a varying number of groups. The values are calculated based on Lemmas 2.1 and 2.2. We see that the local connectivity is extremely far from balanced; even when the nodes are deployed in 128 groups the probability of secure communication between cross-group neighbours is only about $1/9$. This low probability of cross-group key sharing carries with it an inherent risk that entire groups may become disconnected from the network, and may cause bottlenecks in communication between groups, with a small number of nodes forced to bear the brunt of the communication load between groups. This is especially a problem if the number of groups is small.

Additionally, the fact that this scheme imposes 100% connectivity within each group causes problems when the number of groups is small, as it affects the performance of both of the proposed KPSs. The storage requirements of the distinct pairwise keys scheme become prohibitive when the size of each group is large. On the other hand, the resilience of the polynomial-based scheme is reduced if the number of groups is small: when an adversary captures a given number $s \geq t+1$ of nodes, the probability that there is some group to which at least $t+1$ of them belong increases as the number of groups decreases.

This inability to adjust the connectivity means that this scheme is ill-suited to the majority of networks that are deployed in a small number of groups. However, such networks are of arguably the greatest practical interest, as they provide an increased degree of knowledge/control of the nodes' eventual locations, with only a slight increase in the difficulty associated with deploying the nodes. In Section 3.2 we will consider the design of a KPS in which the connectivity can be more readily varied to suit application requirements.

3.2 Flexible Key Predistribution Using Transversal Designs

In Liu, Ning and Du's group-based scheme, the groups and cross-groups form a grid structure, with each node able to establish a key with precisely one node in each other group. Thus when there are a nodes in a group, two nodes from distinct groups have probability $1/a$ of sharing a key. We wish to generalise this grid structure in order to permit schemes where nodes from distinct groups have probability k/a of sharing a key, where k is a parameter that can be varied as desired. We can do this by using a *resolvable transversal design*.

Definition 3.1. [6] *A* transversal design $\mathrm{TD}(k,n)$ *consists of a set $\mathcal{V}$ of cardinality kn whose elements are known as* points*, a partition $\mathcal{G}$ of the points of $\mathcal{V}$ into k sets of size n known as* groups*, and a set $\mathcal{B}$ of k-subsets of $\mathcal{V}$ known as* blocks*, with the property that every pair of distinct points is contained in precisely one element of $\mathcal{G} \cup \mathcal{B}$. A $\mathrm{TD}(k,n)$ is* resolvable *if the blocks can be partitioned into sets $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_s$ such that each point of the design is contained in exactly one block in each set. These sets are known as* parallel classes*.*

It follows directly from the above definition that there are n^2 blocks, k points in each block, and each block intersects each group in precisely one point. (Note that these groups are unrelated to the groups in which the nodes are deployed; where confusion may arise we will refer to them as *design groups* and *node groups* respectively.) There are n parallel classes, each containing n blocks; two parallel blocks do not have any points in common. For all prime powers n, such designs are known to exist for all integers $k \leq n$.

Example 3. As an example, we list the design groups, and the blocks of two of the five parallel classes of a resolvable transversal design $\mathrm{TD}(3,5)$. Such a design has fifteen points; we will label the points by the elements of the set $\{A,B,C,D,E,F,G,H,I,J,K,L,M,N,O\}$.

design group 1	$\{A,B,C,D,E\}$
design group 2	$\{F,G,H,I,J\}$
design group 3	$\{K,L,M,N,O\}$

class 1	$\{A,F,L\}$	class 2	$\{A,I,K\}$
	$\{B,I,N\}$		$\{B,J,M\}$
	$\{C,G,M\}$		$\{C,H,L\}$
	$\{D,H,O\}$		$\{D,F,N\}$
	$\{E,J,K\}$		$\{E,G,O\}$

It can be seen that each point occurs in exactly one group, and that each block has one point from each group (the points within the blocks have been ordered so that the first is from group 1, the second from group 2 and the third from group 3). Furthermore, each point appears once in each of the two parallel classes, and if we consider any block from one of the parallel classes, the points it contains each lie in different blocks in the other class (i.e., any two blocks from the same class are disjoint, and blocks from different classes have at most one point in common).

The use of transversal designs for key predistribution was proposed by Lee and Stinson [16]. In their scheme, the points of a transversal design are associated with keys and the blocks with nodes, with each node storing the keys corresponding to points contained in the associated block. This scheme does not require the design to be resolvable, although the transversal designs TD(k, n) whose construction is described in [16] are resolvable. In our scheme we will similarly associate blocks with nodes and points with keys; however, we will also exploit the resolvability of the design, making use of the parallel classes of blocks to assign nodes to node groups. We will now consider this assignment in detail, as well as the distribution of cross-group keys and in-group keys, before describing our scheme in full.

3.2.1 Group Structure

Consider a network of n groups of nodes, with n nodes in each node group, for some prime power n. If Liu, Ning and Du's group-based KPS is applied to this network, then the nodes are also partitioned into n cross groups with n nodes each, and each cross group has one node in common with each node group. A resolvable TD$(1, n)$, on the other hand, has n parallel classes, each containing n blocks; each block contains one point, and there are n points in total. Each point lies in precisely one block of each parallel class. This implies that we can associate the blocks of the design with the nodes of the network, so that the parallel classes of blocks correspond to node groups. By considering the set of blocks containing a given point to be a cross-group, we have recovered the grid structure used by Liu et al.

By replacing the TD$(1, n)$ by a TD(k, n) for some $1 \leq k \leq n$, we introduce an extra parameter that allows us to increase the cross-group connectivity, at the cost of additional storage; each node is then contained in k cross groups, rather than just one. The structure of the resolvable transversal design implies that each node is contained in precisely k cross groups, each cross group contains at most one node from each group, and two cross groups have at most one node in common.

So far we have only considered the case where the number of groups λ is equal to the square root of the number of nodes in the network n^2. However, we can also accommodate more general networks where λ is any divisor of n, by partitioning the n parallel classes of the TD(k, n) into λ disjoint sets $S_1, S_2, \ldots, S_\lambda$ of n/λ parallel classes and letting the blocks in the parallel classes of each set S_i be associated with the nodes of a node group[2]. When node groups are formed by merging parallel classes in this fashion, each cross group contains more than one node (in fact, precisely n/λ nodes) from each node group. This is necessary to enable connectivity to be maintained without adversely affecting the storage when the number of groups is small. This is because the size of the groups is inversely proportional to the number of groups and, if we were to require each cross group to intersect each node group in precisely one point, then the number of cross groups required to ensure each node can establish keys with a given proportion q of nodes in other groups would grow with the size of the group.

3.2.2 Cross-Group Keys

In the basic version of Lee and Stinson's transversal-design based KPS, a key is associated with each point, and nodes store keys corresponding to the points contained in their associated block. In Scheme 1, the points of the design will be associated with degree t instances of Blom's scheme. Thus all pairs of nodes in a given cross group are able to establish keys, and the storage required is equivalent to $k(t+1)$ keys per node.

3.2.3 In-Group Keys

In Section 3.1 we observed that the connectivity imposed by the techniques suggested by Liu et al. for distributing keys within the groups causes problems when the sizes of the groups are large. In Scheme 1

[2]In fact, we can allow any number $\lambda \leq n$ of node groups by simply discarding the blocks in $n \pmod{\lambda}$ of the parallel classes.

we mitigate these difficulties by employing a separate polynomial-based scheme *in each parallel class*, so that neither storage nor resilience is adversely affected if parallel classes are merged to form large groups. Connectivity within a group is maintained by the fact that in the case where parallel classes are merged, the cross-group keys also provide connectivity between the parallel classes that belong to a given group. A further analysis of connectivity issues will be given in Sections 4.1 and 4.2.

Having motivated the various design elements of Scheme 1, we now describe it in full.

Scheme 1. *Given a prime power n and non-negative integers $1 \leq k \leq n$, $\lambda | n$ and $0 \leq t < {}^{n^2}/_{\lambda} - 1$, we use a TD$(k, n)$ to construct a KPS for a network of n^2 nodes deployed in λ groups $G_1, G_2, \ldots, G_\lambda$ of size ${}^{n^2}/_{\lambda}$ as follows:*

1. *Partition the parallel classes $P_1, P_2, \ldots, P_n$ of the TD(k, n) into λ sets of ${}^{n}/_{\lambda}$ parallel classes, denoted $S_1, S_2, \ldots, S_\lambda$. Each S_i contains ${}^{n^2}/_{\lambda}$ blocks.*
2. *Associate each node with a block of the design, such that the nodes of group G_i correspond to the blocks in the parallel classes contained in set S_i.*
3. *Associate a degree t Blom's scheme with each point of the design, and assign shares in a given scheme to each node whose corresponding block contains the relevant point. (Note that resilience is maximised by taking $t = {}^{n^2}/_{\lambda} - 2$, which is equivalent in terms of storage and resilience to assigning a distinct key to each pair of nodes involved in the scheme. As such it is never necessary to consider $t \geq {}^{n^2}/_{\lambda} - 1$.)*
4. *Associate a degree t Blom's scheme with each parallel class P_i, for $1 \leq i \leq n$, and assign shares in a given scheme to each node whose corresponding block is contained within that parallel class.*

This scheme is essentially an instance of the multiple space scheme described by Lee and Stinson [16], but with the resolvability of the transversal design used to align the pattern of key sharing with the group structure of the network. It inherits the beneficial properties of this scheme; in particular, shared key discovery can be carried out without incurring communication overheads (see [16] for details).

Example 4. Suppose we use the TD$(3, 5)$ of Example 3 and take $t = 1$ in Scheme 1 to construct a KPS for 25 nodes deployed in 5 groups of 5 nodes, with each node storing the equivalent of 8 keys.

In this case, ${}^{n}/_{\lambda} = 1$, so each of the sets S_i for $i = 1, 2, \ldots, 5$ consists of a single parallel class of the design. Suppose the nodes in the first group have IDs l, m, r, s, t and the nodes in the second group have IDs u, v, w, x, y. If we denote the share that node l holds in the Blom's scheme associated with the point A by f_l^A, and the share that the node l holds in the Blom's scheme associated with the i^{th} parallel class by f_l^i and so on, then the first two node groups consist of nodes possessing the following sets of shares:

$$
\begin{array}{llll|llll}
\text{group 1} & l: & \{f_l^A, f_l^F, f_l^L, f_l^1\} & & \text{group 2} & u: & \{f_u^A, f_u^I, f_u^K, f_u^2\} \\
& m: & \{f_m^B, f_m^I, f^N, f_m^1\} & & & v: & \{f_v^B, f_v^J, f_v^M, f_v^2\} \\
& r: & \{f_r^C, f_r^G, f_r^M, f_r^1\} & & & w: & \{f_w^C, f_w^H, f_w^L, f_w^2\} \\
& s: & \{f_s^D, f_s^H, f_s^O, f_s^1\} & & & x: & \{f_x^D, f_x^F, f_x^N, f_x^2\} \\
& t: & \{f_t^E, f_t^J, f_t^K, f_t^1\} & & & y: & \{f_y^E, f_y^G, f_y^O, f_y^2\}
\end{array}
$$

Theorem 3.2. *The KPSs constructed in Scheme 1 have the following properties:*

1. *Each node stores the equivalent of $(k + 1)(t + 1)$ keys.*
2. *For each instance of Blom's scheme, there are n nodes possessing shares in that scheme.*
3. $q = \frac{k}{n}$
4. $p = \frac{n-1}{\frac{n^2}{\lambda}-1} + \frac{(\frac{n^2}{\lambda}-n)}{\frac{n^2}{\lambda}-1}\frac{k}{n}$
5. $$\mathsf{fail}(s) = \begin{cases} 0 & s \leq t, \\ 1 - \displaystyle\sum_{i=0}^{t} \frac{\binom{n-2}{i}\binom{n^2-n}{s-i}}{\binom{n^2-2}{s}} & s > t. \end{cases} \tag{3}$$

Proof. (1) Each block of a transversal design TD(k, n) contains k points, and is contained within one parallel class. Each node thus stores shares in $k+1$ Blom's schemes, which requires equivalent storage to $(k+1)(t+1)$ keys.

(2) Each point of a TD(k, n) is contained in precisely n blocks, and each parallel class contains precisely n blocks, hence shares in each Blom's scheme are allocated to n nodes.

(3) Two nodes from different node groups possess shares in a common Blom's scheme if their corresponding blocks intersect. As any given block contains k points of the design, each of which lie in $n-1$ further blocks, this happens with probability ${}^{k(n-1)}/_{(n^2-n)} = {}^{k}/_{n} = q$.

(4) Two nodes from the same group correspond to blocks from the same parallel class with probability ${}^{(n-1)}/_{({}^{n^2}/_{\lambda}-1)}$, in which case they both possess shares in a common Blom's scheme. Otherwise, with probability ${}^{({}^{n^2}/_{\lambda}-n)}/_{({}^{n^2}/_{\lambda}-1)}$ their blocks are in different parallel classes, in which case they possess shares in a common scheme with probability ${}^{k}/_{n}$. Hence the overall probability that two nodes selected at random from the same group share a key is

$$p = \frac{n-1}{\frac{n^2}{\lambda}-1} + \frac{(\frac{n^2}{\lambda}-n)}{\frac{n^2}{\lambda}-1}\frac{k}{n}. \tag{4}$$

We note that, for $\lambda > 1$, we have $q < p$.

(5) Each link, whether between nodes within a group, or from different groups, arises from two nodes possessing shares in a common Blom's scheme, in which $n-2$ further nodes possess shares. Such a link fails if at least $t+1$ nodes with shares in that scheme are captured; as nodes are captured without replacement, this occurs with the probability given in Equation 3. □

Thus we see that the storage and resilience of Scheme 1 depend on the values of the parameters λ, k, and t, and the connectivity depends on k and λ. In Section 4 we will see how the properties of the KPS vary as these parameters are altered.

4 Analysis of the Performance of Group-Based KPSs

In Section 3.2 we described the construction of a KPS for use in a group-deployed environment (Scheme 1), and discussed some of its basic properties (Theorem 3.2). We now proceed to give a more detailed analysis of the behaviour of this scheme when it is used for key distribution in a group-deployed network. In Section 4.1 we demonstrate the beneficial effect of group deployment on the network connectivity, by considering the use of Scheme 1 when the number of groups varies, but the other parameters are fixed. In Section 4.2 we show that Scheme 1 outperforms other schemes in the literature for various combinations of resilience, storage, and connectivity.

4.1 Improving Connectivity through the Use of Group Deployment

We consider a network of 16384 nodes whose keys are distributed according to Scheme 1, based on a TD$(39, 128)$ and taking $t = 0$, and assuming that $\eta = 7$. In this KPS, the number of keys stored by each node $(k+1)$, and the number of nodes that share each key (and hence the value of $\mathsf{fail}(s)$) do not depend on the value of λ that is chosen, hence we can investigate how the connectivity varies when the storage and resilience are fixed. Combining the values of p and q given by Theorem 3.2 with the formulas in Lemmas 2.1 and 2.2 enables us to express the local connectivity in terms of the number of groups in which the nodes are deployed. Figure 5 shows the resulting probabilities when a TD$(39, 128)$ is used to distribute keys to a network of 16384 nodes with $\eta = 7$.

We see that the local connectivity probabilities increase with the number of groups, and that there is a certain number of groups for which the local connectivity is most closely balanced (for the parameters used in Figure 5 this number is 32). If the number of groups is increased beyond this point, however, both in-group and cross-group probabilities continue to increase, and the difference between them is not too large. Hence, from a connectivity point of view, it is better to choose a larger number of groups, although this must be traded off against the potential practical difficulties involved in deploying a larger number of groups.

λ	in-group	cross-group
1	0.660	0.826
2	0.671	0.831
4	0.694	0.839
8	0.736	0.855
16	0.809	0.882
32	0.914	0.923
64	0.993	0.969
128	1	0.996

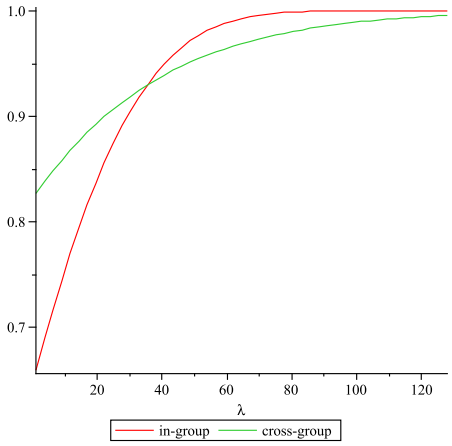


Figure 5: The probability that neighbouring nodes from within a group or from different groups can communicate securely using a KPS based on a TD$(39, 128)$ in a network of 16384 nodes deployed in λ groups, with $\eta = 7$.

4.2 Varying Connectivity and Resilience

Having seen how a varying number of groups affects the Scheme 1's connectivity, we now wish to examine further the relationship between the resilience and connectivity of this KPS, and of other schemes in the literature. We assume a network of 16384 nodes deployed in 16 groups of 1024 nodes each. In order to ensure a fair comparison between schemes, we consider several fixed values of the number of keys stored by each node (m), and the expected proportion of pairs of nodes that can establish a shared key, which we denote by Pr_1. We observe that Pr_1 is equal to the expected number of edges in the block graph, divided by the number of distinct pairs of nodes in the network. For each combination of these parameters, we explore the connectivity of the KPS of Liu et al. (LND), Scheme 1, and Eschenauer and Gligor's KPS (EG) by computing the in-group and cross-group key sharing probabilities p and q, and use the formulas from Lemmas 2.1 and 2.2 to estimate the probability that neighbouring nodes from the same group (resp., different groups) can communicate securely by either a one-hop or a two-hop path, which we denote by $\mathrm{2hop}_i$ and $\mathrm{2hop}_c$. For Scheme 1 based on a TD(k, n), we can compute Pr_1 for a network of n^2 nodes deployed in λ groups by

$$\mathrm{Pr}_1 = \frac{(k+1)n\binom{n}{2}}{\binom{n^2}{2}}. \tag{5}$$

For LND we have

$$\mathrm{Pr}_1 = \frac{\lambda\binom{n^2/\lambda}{2} + \frac{n^2}{\lambda}\binom{\lambda}{2}}{\binom{n^2}{2}}, \tag{6}$$

and for EG with a key pool of size K we have

$$\mathrm{Pr}_1 = \frac{\binom{K-m}{m}}{\binom{K}{m}}. \tag{7}$$

To provide a measure of global connectivity, we simulate the deployment of the nodes in a $500m \times 500m$ target region in which the nodes of each group are deployed uniformly within a circle of radius $100m$, with these circles being placed uniformly within the target region. Assuming that the nodes have a communication range of $6m$ we then estimate the mean size of the largest connected component of the network ($E(M)$), based on 100 trials. The results of these computations are shown in Table 1.

We examine the resilience of each scheme for various sets of parameters by plotting the value of $\mathsf{fail}(s)$ as s varies from 0 to 1000. The expression for $\mathsf{fail}(s)$ for Scheme 1 is given in Equation (3). When LND using polynomials of degree t is applied to a network of n^2 nodes deployed in λ groups, there are $\lambda\binom{n^2/\lambda}{2}$

Scheme	parameters	m	$\mathbf{Pr}_1$	p	q	$\mathbf{2hop}_i$	$\mathbf{2hop}_c$	$\mathbf{E}(M)$
LND	$t = 17$	32	0.0634	1.00	0.000977	1.00	0.0145	1510
Scheme 1	$k = 7$,$t = 3$	32	0.0620	0.172	0.0547	0.329	0.172	2140
EG	$K = 15760$	32	0.0630	0.0630	0.0630	0.0888	0.114	1800
Scheme 1	$k = 15$,$t = 1$	32	0.124	0.227	0.117	0.466	0.395	8640
EG	$K = 7750$	32	0.124	0.124	0.124	0.215	0.296	7600
Scheme 1	$k = 31$,$t = 0$	32	0.248	0.336	0.242	0.713	0.769	13200
EG	$K = 3620$	32	0.248	0.248	0.248	0.518	0.691	12600
Scheme 1	$k = 63$,$t = 0$	64	0.496	0.555	0.492	0.966	0.994	15000
EG	$K = 6050$	64	0.496	0.496	0.496	0.930	0.990	14800

Table 1: Measures of the connectivity of KPSs for a network of 16384 nodes deployed in 16 groups. All values are given to three significant figures (other than parameter sizes, which are exact).

in-group links that fail with probability

$$p_1 = 1 - \sum_{i=0}^{t} \frac{\binom{n^2/\lambda - 2}{i}\binom{n^2 - n^2/\lambda}{s-i}}{\binom{n^2-2}{s}} \tag{8}$$

after the capture of $s > t$ nodes, and $\frac{n^2}{\lambda}\binom{\lambda}{2}$ cross-group links that fail with probability

$$p_2 = 1 - \sum_{i=0}^{t} \frac{\binom{\lambda-2}{i}\binom{n^2-\lambda}{s-i}}{\binom{n^2-2}{s}} \tag{9}$$

after the capture of $s > t$ nodes. Hence $\mathsf{fail}(s)$ for this scheme is given by

$$\mathsf{fail}(s) = \begin{cases} 0 & s \le t, \\ \frac{n^2-\lambda}{n^2-2\lambda+\lambda^2} p_1 + \frac{\lambda(\lambda-1)}{n^2-2\lambda+\lambda^2} p_2 & s > t. \end{cases} \tag{10}$$

In the case of EG, if n^2 nodes each store m keys from a pool of size K, then each key is shared by an average of $n^2 m/K$ nodes. Thus for this scheme, we have

$$\mathsf{fail}(s) \approx 1 - \frac{\binom{n^2 - n^2 m/K}{s}}{\binom{n^2-2}{s}}. \tag{11}$$

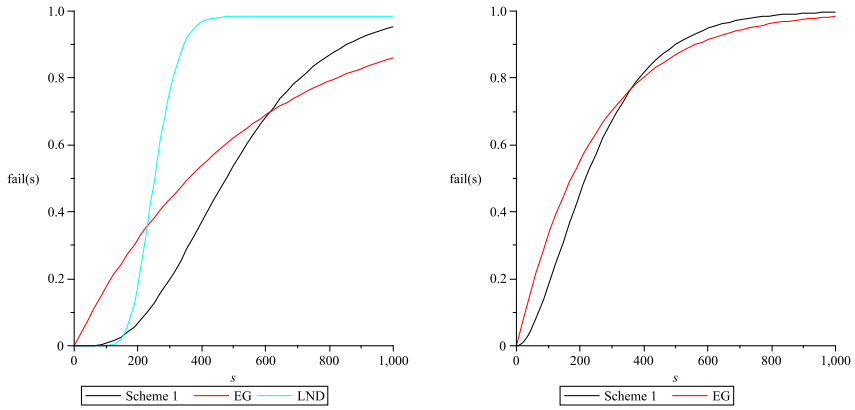
These expressions are plotted for various parameters in Figures 6(a), 6(b) and 6(c).

4.2.1 Discussion of Connectivity Results

When the value of Pr_1 is fixed the schemes have the same expected number of links, but their differing p and q values reflect a difference in how those links are distributed relative to the group structure of the network. Table 1 shows that for each combination of m and Pr_1, Scheme 1 achieves higher values of $\mathrm{2hop}_i$ and $\mathrm{2hop}_c$ than EG, and for $\mathrm{Pr}_1 \approx 0.063$ the two-hop values it obtains are significantly more balanced than those of LND. This suggests that the pattern of connectivity achieved by Scheme 1 is better suited to the properties of a network with group deployment. The simulated networks had larger connected components on average when Scheme 1 was used than when either of the other KPSs were employed, reinforcing this conclusion.

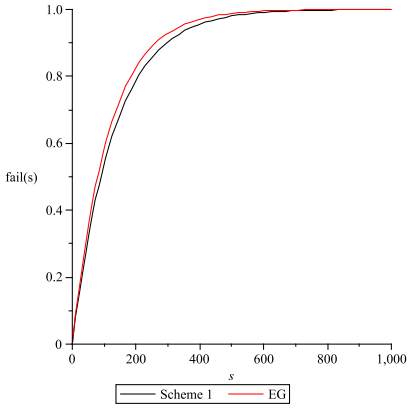
For a network of 16384 nodes in 16 groups, LND is constrained to $\mathrm{Pr}_1 = 0.0634$. For this value of Pr_1, each scheme gives rise to relatively small connected components, indicating that this value of Pr_1 is simply too low to permit these schemes to achieve adequate global connectivity for these network parameters. In particular, this suggests that LND is not a suitable KPS for use in this specific group-deployed environment.

When parameters are chosen for Scheme 1 or EG to allow higher values of Pr_1, the observed sizes of the largest connected components increase accordingly. In fact, restricting m to be 32, or even 64,



(a) $\Pr_1 \approx 0.063$ $m = 32$

(b) $\Pr_1 = 0.124$ $m = 32$



(c) $\Pr_1 = 0.248$ $m = 32$

Figure 6: An illustration of $\mathsf{fail}(s)$ for Scheme 1, EG and LND with different values of $\Pr_1$.

is relatively conservative. With larger storage, (or, for example, if the nodes were to have a larger communication range) we would expected to be able to obtain higher levels of connectivity. Similarly, while the values of 2hop_i and 2hop_c can be observed to become more balanced as $\Pr_1$ increases for both Scheme 1 and EG, the precise extent of the balancing will vary if different values of η are considered.

4.2.2 Discussion of Resilience

We see in Figure 6(a) (where $\Pr_1 \approx 0.063$ and $m = 32$) that LND has the lowest value of $\mathsf{fail}(s)$ when s is small (due to the comparatively high degree of the polynomials it uses), but that $\mathsf{fail}(s)$ rapidly approaches 1 as s increases (due to the fact that large numbers of nodes have shares in each in-group polynomial.) The value of $\mathsf{fail}(s)$ for Scheme 1 is slightly higher until $s \approx 200$, after which it is substantially lower, and it does not exceed the value obtained by EG until s is greater than 500. As the connectivity increases (Figures 6(b), 6(c)), the behaviour of $\mathsf{fail}(s)$ for Scheme 1 gradually approaches that of EG. This is because the fixed storage is forcing the parameter t of Scheme 1 to decrease; in the case where $t = 0$ we would expect the properties of Scheme 1 and EG to be very similar, as reflected by Figure 6(c). For given $\Pr_1$ and m, if $t > 0$ we would expect $\mathsf{fail}(s)$ to be lower for Scheme 1 when s is not too large; this is arguably the preferred behaviour. When combined with the fact that Scheme 1 can also achieve greater two-hop connectivity and better global connectivity for such parameters, this provides additional evidence of the benefits that can be obtained by exploiting group deployment of a network.

5 Conclusion

We have seen that the use of group deployment can lead to better connectivity outcomes for a network, but that it is necessary to be able to adapt the connectivity and resilience properties of a KPS in order to fully exploit this advantage. The analysis of Section 4.2 has shown that the Scheme 1 proposed in Section 3.2 displays good resilience and connectivity in a group deployed environment, over a range of different parameters. To summarise:

- The scheme is flexible, as it is possible to vary the number of nodes, number of groups, and the storage requirements over a practical range of parameters, and to trade connectivity and resilience against the storage requirements.
- The fact that $p > q$ not only leads to more balanced connectivity, it also increases the total number of secure links between neighbouring nodes after deployment, since pairs of nodes from within a group are more likely to be neighbours than nodes from different groups.
- Being a deterministic KPS, this scheme enjoys the corresponding benefits such as efficient shared-key discovery with no communication overheads (see [16] for details.) The use of a transversal design leads to schemes with nice regular properties: two nodes share at most one key and the number of nodes sharing a key is constant, as is the number of keys stored by each node.

Several generalisations of Scheme 1 are possible: for instance, the Blom's schemes could be replaced with other types of KPS (such as that of Eschenauer and Gligor). It is an open question whether this could provide better results in specific network environments.

References

[1] R. Blom, An optimal class of symmetric key generation systems, in: T. Beth, N. Cot, I. Ingemarsson (eds.), Advances in Cryptology -EUROCRYPT '84, vol. 209 of LNCS, Springer-Verlag, 1985, pp. 335–338.

[2] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung, Perfectly-secure key distribution for dynamic conferences., in: E. F. Brickell (ed.), Advances in Cryptology -CRYPTO '92, vol. 740 of LNCS, Springer-Verlag, 1992, pp. 471–486.

[3] N. T. Canh, Y.-K. Lee, S. Lee, HGKM: A group-based key management scheme for sensor networks using deployment knowledge, in: CNSR, IEEE Computer Society, Los Alamitos, CA, USA, 2008, pp. 544–551.

[4] S. A. Çamtepe, B. Yener, Key distribution mechanisms for wireless sensor networks: a survey, Tech. Rep. TR-05-07, Rensselaer Polytechnic Institute (March 2005).

[5] J. Y. Chun, Y. H. Kim, J. Lim, D. H. Lee, Location-aware random pair-wise keys scheme for wireless sensor networks, Security, Privacy and Trust in Pervasive and Ubiquitous Computing, 2007. SECPerU 2007. Third International Workshop on (2007) 31–36.

[6] C. Colbourn, J. Dinitz (eds.), The CRC Handbook of Combinatorial Designs, CRC Press, Boca Raton, 1996.

[7] H. Dai, H. Xu, An efficient key predistribution scheme in wireless sensor networks, in: IEEE ISWCS 2007, 4th International Symposium on Wireless Communication Systems, 2007, pp. 31–34.

[8] A. K. Das, I. Sengupta, An effective group-based key establishment scheme for large-scale wireless sensor networks using bivariate polynomials, Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on (2008) 9–16.

[9] F. Delgosha, E. Ayday, F. Fekri, MKPS: a multivariate polynomial scheme for symmetric key-establishment in distributed sensor networks, in: M. Guizani, H.-H. Chen, X. Zhang (eds.), IWCMC, ACM, 2007, pp. 236–241.

[10] F. Delgosha, F. Fekri, Key pre-distribution in wireless sensor networks using multivariate polynomials, in: IEEE Commun. Soc. Conf. Sensor and Ad Hoc Commun. and Networks - SECON05, 2005.

[11] F. Delgosha, F. Fekri, Threshold key-establishment in distributed sensor networks using a multivariate scheme, in: INFOCOM, IEEE, 2006.

[12] W. Du, J. Deng, Y. S. Han, S. Chen, P. K. Varshney, A key management scheme for wireless sensor networks using deployment knowledge, in: INFOCOM 2004, p. 597.

[13] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks, in: CCS '02: Proceedings of the 9th ACM conference on Computer and communications security, ACM Press, New York, NY, USA, 2002, pp. 41–47.

[14] D. Huang, M. Mehta, D. Medhi, L. Harn, Location-aware key management scheme for wireless sensor networks, in: SASN '04: Proceedings of the 2nd ACM workshop on Security of Ad hoc and Sensor Networks, ACM, New York, NY, USA, 2004, pp. 29–42.

[15] P. Kotzanikolaou, D. D. Vergados, G. Stergiou, E. Magkos, Multilayer key establishment for large-scale sensor networks, Int. J. Secur. Netw. 3 (1) (2008) 1–9.

[16] J. Lee, D. R. Stinson, On the construction of practical key predistribution schemes for distributed sensor networks using combinatorial designs, ACM Trans. Inf. Syst. Secur. 11 (2) (2008) 1–35.

[17] G. Li, J. He, Y. Fu, A hexagon-based key predistribution scheme in sensor networks, in: ICPPW '06: Proceedings of the 2006 International Conference Workshops on Parallel Processing, IEEE Computer Society, Washington, DC, USA, 2006, pp. 175–180.

[18] G. Li, J. He, Y. Fu, Key predistribution in sensor networks, in: J. Ma, H. Jin, L. T. Yang, J. J. P. Tsai (eds.), UIC, vol. 4159 of LNCS, Springer-Verlag, 2006, pp. 845–853.

[19] D. Liu, P. Ning, Location-based pairwise key establishments for static sensor networks, in: S. Setia, V. Swarup (eds.), SASN, ACM, 2003, pp. 72–82.

[20] D. Liu, P. Ning, W. Du, Group-based key pre-distribution in wireless sensor networks, in: WiSe '05: Proceedings of the 4th ACM workshop on Wireless Security, 2005, pp. 11–20.

[21] D. Liu, P. Ning, W. Du, Group-based key predistribution for wireless sensor networks, ACM Trans. Sen. Netw. 4 (2) (2008) 1–30.

[22] Y. Mao, M. Wu, Coordinated sensor deployment for improving secure communications and sensing coverage, in: SASN '05: Proceedings of the 3rd ACM workshop on Security of Ad hoc and Sensor Networks, ACM, New York, NY, USA, 2005, pp. 117–128.

[23] K. M. Martin, M. Paterson, An application-oriented framework for wireless sensor network key establishment, Electron. Notes Theor. Comput. Sci. 192 (2) (2008) 31–41.

[24] M. Ren, J. Jaworski, K. Rybarczyk, Random key predistribution for wireless sensor networks using deployment knowledge, 8th Central European Conference on Cryptography (2008).

[25] K. Römer, F. Mattern, The design space of wireless sensor networks, IEEE Wireless Communications Magazine 11 (6) (2004) 54–61.

[26] K. Simonova, A. C. H. Ling, X. S. Wang, Location-aware key predistribution scheme for wide area wireless sensor networks, in: SASN '06: Proceedings of the fourth ACM workshop on Security of Ad hoc and Sensor Networks, ACM, New York, NY, USA, 2006, pp. 157–168.

[27] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, M. Galloway, A survey of key management schemes in wireless sensor networks, Comput. Commun. 30 (11-12) (2007) 2314–2341.

[28] C. Yang, J. Xiao, Location-based pairwise key establishment and data authentication for wireless sensor networks, Information Assurance Workshop, 2006 IEEE (2006) 247–252.

[29] L. Zhou, J. Ni, C. V. Ravishankar, Efficient key establishment for group-based wireless sensor deployments, in: WiSe '05: Proceedings of the 4th ACM workshop on Wireless Security, ACM, New York, NY, USA, 2005, pp. 1–10.

[30] Y. Zhou, Y. Zhang, Y. Fang, Key establishment in sensor networks based on triangle grid deployment model, Military Communications Conference, 2005. MILCOM 2005. IEEE (17-20 Oct. 2005) 1450–1455 Vol. 3.

[31] Y. Zhou, Y. Zhang, Y. Fang, LLK: a link-layer key establishment scheme for wireless sensor networks, Wireless Communications and Networking Conference, 2005 IEEE 4 (2005) 1921–1926 Vol. 4.