Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks

Seyit A. Çamtepe and Bülent Yener

Department of Computer Science, Rensselaer Polytechnic Institute,
Troy, NY 12180, USA
{camtes,yener}@cs.rpi.edu

Abstract. Key distribution is one of the most challenging security issues in wireless sensor networks where sensor nodes are randomly scattered over a hostile territory. In such a sensor deployment scenario, there will be no prior knowledge of post deployment configuration. For security solutions requiring pairwise keys, it is impossible to decide how to distribute key pairs to sensor nodes before the deployment. Existing approaches to this problem are to assign more than one key, namely a key-chain, to each node. Key-chains are randomly drawn from a key-pool. Either two neighboring nodes have a key in common in their key-chains, or there is a path, called key-path, among these two nodes where each pair of neighboring nodes on this path has a key in common. Problem in such a solution is to decide on the key-chain size and key-pool size so that every pair of nodes can establish a session key directly or through a path with high probability. The size of the key-path is the key factor for the efficiency of the design. This paper presents novel, *deterministic* and *hybrid* approaches based on *Combinatorial Design* for key distribution. In particular, several *block design* techniques are considered for generating the key-chains and the key-pools.
Comparison to probabilistic schemes shows that our combinatorial approach produces better connectivity with smaller key-chain sizes.

1 Introduction and Problem Definition

In this work, we consider a sensor network in which sensor nodes need to communicate with each other for data processing and routing. We assume that the sensor nodes are distributed to the target area in large numbers and their location within this area is determined randomly. These type of sensor networks are typically deployed in adversarial environments such as military applications where a large number of sensors may be dropped from airplanes.

In this application, secure communication among sensor nodes requires authentication, privacy and integrity. In order to establish this, there must be a *secret key* shared between a pair of communicating sensor nodes. Because the network topology is unknown prior to deployment, a key pre-distribution scheme is required where keys are stored into ROMs of sensors before the deployment. The keys stored must be carefully selected so to increase the probability that

P. Samarati et al. (Eds.): ESORICS 2004, LNCS 3193, pp. 293–308, 2004.

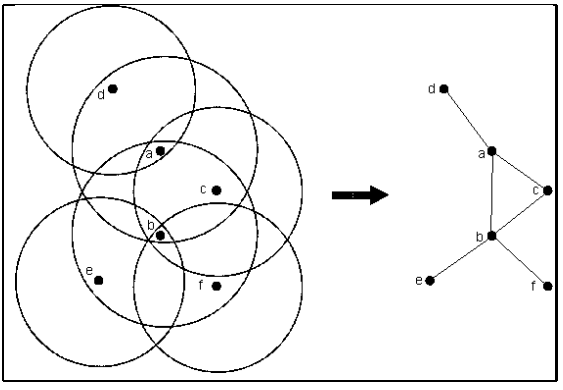

Fig. 1. A Wireless Sensor Network

two neighboring sensor nodes have at least one key in common. Nodes that do not share a key directly may use a path where each pair of nodes on the path shares a key. The length of this path is called *key-path* length. Average *key-path* length, is an important performance metric and design consideration. Consider sample sensor network given in Figure-1. Assume that only sensor nodes a and b does not share a key. Nodes a and c can establish secure communication where the key-path length is 1. Node c and b also have key-path length of one. However, nodes a and b can only use the path a-c-b to communicate securely with key-path length of two.

The common approach is to assign each sensor node multiple keys, *randomly* drawn from a *key-pool*, to construct a *key-chain* to ensure that either two neighboring nodes have a key in common in their key-chain, or there is a key-path. Thus the challenge is to decide on the key-chain size and key-pool size so that every pair of nodes can establish a session key directly or through a path. Key-chain size is limited by the storage capacity of the sensor nodes. Moreover, very small key-pool increases the probability of key share between any pair of sensor nodes by decreasing the security in that, the number of the keys needed to be discovered by the adversary decreases. Similarly, very large key-pool decreases the probability of key share by increasing the security.

Eschenauer *et al.* in [14] propose a *random key pre-distribution scheme* where tens to hundreds of keys are uploaded to sensors before the deployment. In their solution, initially a large key pool of P and the key identities are generated. For each sensor, k keys are randomly drawn from the key-pool P without replacement. These k keys and their identities form a *key-chain* which is loaded in to the memory of the sensor node. Two neighboring nodes compare list of identities of keys in their key-chain. Since only the identities are exchanged, this process can take place without any privacy mechanism. Eschenauer *et al.* also propose to employ *Merkle Puzzle* [20] similar approach to secure key identities. After key identity exchange, common key(s) are used to secure the link in between two sensor nodes. It may be the case that some of the neighboring nodes may not be

able to find a key in common. These nodes may communicate securely through other nodes, through other secured links. Chan *et al.* in [6] propose a modification to the basic scheme of Eschenauer *et al.* They increase the amount of key overlap required for key-setup. That is, q common keys are needed instead of one to be able to increase the security of the communication between two neighboring nodes. In [33], common keys in the key-chains are used to establish multiple logical paths over which *threshold key sharing scheme* is used to agree on a new secret.

Random-pairwise key scheme in [6] is a modification of the pairwise key scheme. It is based on *Erdos and Renyi's* work; to achieve probability p of any two nodes are connected, in a network of n nodes, each node needs to store only a random set of np pairwise keys instead of $n-1$. Slijepcevic *et al.* [24] propose that each sensor node shares a list of master keys, a random function and a seed. Every sensor uses shared random function and shared seed to select a network wise or group wise master key. In [3, 18], polynomial-based key pre-distribution protocol proposed for group key pre-distribution. In [19], polynomial pool-based key pre-distribution is used for pairwise key establishment. For each sensor, random or a grid based pre-distribution scheme is used to select set of polynomials from a pool.

In [2], Blom proposes a λ-secure key pre-distribution system where a public matrix P and a private symmetric matrix S over a finite field $GF(q)$ is used. Rows of the matrix $A = (S.P)^T$ is distributed to users. Blom's scheme is a deterministic scheme where any pair of nodes can find a common secret key. Du *et al.* in [9] use Blom's scheme with multiple spaces to increase resilience. In [10], Du *et al.* first model node deployment knowledge in a wireless sensor network and then develop a key pre-distribution scheme based on this model.

In [23, 30, 11, 12, 7, 5, 34] a network architecture where there are one or more base-stations is considered. These base-stations are considered as powerful in resource and sensor nodes are clustered around them. Each sensor node shares a key with each base-station to secure sensor node to base-station and base-station to sensor node unicast communication. Authentication mechanism for the broadcasts from base-station to sensor nodes is addressed in [23, 11, 12, 17, 7]. They propose modified versions of *TESLA* where a verifiable key, which is used to encrypt a message, is disclosed later then the message broadcasted.

1.1 Our Contributions and Organization of This Work

The main contribution of this work is the *deterministic* and *hybrid* approaches to the key distribution problem. In particular, we bring in a novel construction methodology from *Combinatorial Design Theory* to address this problem. Although there are some applications of Combinatorial Designs in cryptography [26–28], and in network design [32, 29], best to our knowledge this work is the first to apply design theory to key distribution. Our analysis indicate that deterministic approach has strong advantages over the randomized ones since it (i) increases the probability that two nodes will share a key, and (ii) decreases the key-path length.

This paper is organized as follows: In Section 2 we provide a brief background to the combinatorial designs used in this work without exceeding the scope of this paper. In Section 3 we introduce our key distribution construction and explain the mapping from design theory to this practical problem. In Section 4 we address scalability issues. In Section 5, we present our analysis and comparison with randomized methods. Finally, in Section 6 we conclude.

2 Background on Combinatorial Designs

A *Balanced Incomplete Block Design (BIBD)* is an arrangement of v distinct objects into b blocks such that each block contains exactly k distinct objects, each object occurs in exactly r different blocks, and every pair of distinct objects occurs together in exactly λ blocks. The design can be expressed as (v, k, λ), or equivalently (v, b, r, k, λ), where: $\lambda\ (v-1)\ =\ r\ (k-1)$ and $b\ k\ =\ v\ r$.

2.1 Symmetric BIBD

A *BIBD* is called *Symmetric BIBD* or *Symmetric Design* when $b = v$ and therefore $r = k$ [8, 1, 15, 31]. A *Symmetric Design* has four properties: every block contains $k = r$ elements, every element occurs in $r = k$ blocks, every pair of elements occurs in λ blocks and every pair of blocks intersects in λ elements.

In this paper, we are interested in a subset of Symmetric Designs, called a *Finite Projective Plane*. A *Finite Projective Plane* consists of a finite set P of points and a set of subsets of P, called lines. For an integer n where $n \geq 2$, *Finite Projective Plane* of order n has four properties: (i) every line contains exactly $n+1$ points, (ii) every point occurs on exactly $n+1$ lines, (iii) there are exactly n^2+n+1 points, and (iv) there are exactly n^2+n+1 lines. If we consider lines as blocks and points as objects, then a *Finite Projective Plane* of order n is a *Symmetric Design* with parameters $(n^2+n+1, n+1, 1)$ [8, 1].

Given a block design $D = (v, k, \lambda)$ with a set S of $|S| = v$ objects and $B = \{B_1, B_2, \ldots, B_b\}$ of $|B| = b$ blocks where each block includes exactly k objects, *Complementary Design* $\overline{D}$ has the complement blocks $\overline{B_i} = S - B_i$ as its blocks for $1 \leq i \leq b$. $\overline{D}$ is a block design with parameters $(v, b, b-r, v-k, b-2r+\lambda)$ where $(b-2r+\lambda > 0)$ [1, Theorem 1.1.6]. If $D = (v, k, \lambda)$ is a *Symmetric Design*, then $\overline{D} = (v, v-k, v-2r+\lambda)$ is also a *Symmetric Design* [1, Corollary 1.1.7].

2.2 Finite Generalized Quadrangle

A *Finite Generalized Quadrangle (GQ)* is an incidence structure $S = (P, B, I)$ where P and B are disjoint and nonempty sets of points and lines respectively, and for which I is a symmetric point-line incidence relation satisfying the following axiom:

1. Each point is incident with $t+1$ lines ($t \geq 1$) and two distinct points are incident at most one line,

2. Each line is incident with $s+1$ points ($s \geq 1$) and two distinct lines are incident with at most one point,
3. If x is a point and L is a line not incident (I) with x, then there is a unique pair $(y, M) \in PXB$ for which $x\ I\ M\ I\ y\ I\ L$.

In this work, we are interested in three known GQ's as defined in [21, 13, 16, 22]: two GQs are from the *Projective Space* $PG(4,q)$ and $PG(5,q)$ of order q, third one is from $PG(4,q^2)$ of order q^2. Let function f be an *irreducible binary quadratic*, then the three GQs can be defined as follows:

1. $GQ(s,t) = GQ(q,q)$ from $PG(4,q)$ with canonical equation $x_0^2 + x_1x_2 + x_3x_4 = 0$: $GQ(q,q) \Rightarrow s = t = q$, $v = b = (q+1)(q^2+1)$.
2. $GQ(s,t) = GQ(q,q^2)$ from $PG(5,q)$ with canonical equation $f(x_0,x_1) + x_2x_3 + x_4x_5 = 0$: $GQ(q,q^2) \Rightarrow s = q$, $t = q^2$, $v = (q+1)(q^3+1)$, $b = (q^2+1)(q^3+1)$.
3. $GQ(s,t) = GQ(q^2,q^3)$ from $PG(4,q^2)$ with canonical equation $x_0^{q+1} + x_1^{q+1} + ... + x_d^{q+1} = 0$: GQ(q^2,q^3) $\Rightarrow s = q^2$, $t = q^3$, $v = (q^2+1)(q^5+1)$, $b = (q^3+1)(q^5+1)$.

Consider $GQ(s,t) = GQ(q,q)$ in which lines are mapped to blocks and points to objects. Thus, there are $v = b = (q+1)(q^2+1)$ blocks and objects where each block contains $s+1 = q+1$ objects and where each object is contained in $t+1 = q+1$ blocks.

3 Combinatorial Design to Key Distribution

In the following two sections, we describe how *Symmetric Designs* and *Generalized Quadrangles* are used to generate key-chains for the sensors in a sensor network.

3.1 Mapping from Symmetric Design to Key Distribution

In this work, we are interested in *Finite Projective Plane* of order n which is a *Symmetric Design (Symmetric BIBD)* with parameters $(n^2+n+1, n+1, 1)$.

Mapping: We assume a distributed sensor network where there are N sensor nodes. Sensor nodes communicate with each other and require pairwise keys to secure their communication. Each sensor has a *key-chain* of K keys which is stored to its ROM before the deployment. Keys are selected from a set P of *key-pool*. To secure the communication between them, a pair of sensor nodes need to have χ keys in common in their key-chain. Based on this, we define mapping given in Table-1

For a sensor network of N nodes, with total of N key-chains, a Symmetric Design with $b \geq N$ blocks needs to be constructed by using set S with $|S| = v = b$ objects. That means, $b = v = n^2+n+1 \geq N$ for a prime power n [8, 1]. Each object in S can be associated with a distinct random key, and each block can be

Table 1. Mapping from Symmetric Design to Key Distribution

Symmetric Design		Key Distribution
Object Set (S)	$\rightarrow$	Key-Pool (P)
Object Set Size ($\|S\| = v = n^2 + n + 1$)	$\rightarrow$	Key-Pool Size ($\|P\|$)
Blocks	$\rightarrow$	Key-Chains
# Blocks ($b = n^2 + n + 1$)	$\rightarrow$	# Key-Chains (N)
# Blocks ($b = n^2 + n + 1$)	$\rightarrow$	# Sensor Nodes (N)
# Objects in a Block ($k = n + 1$)	$\rightarrow$	# Keys in a Key-Chain (K)
# Blocks that an Object is in ($r = n + 1$)	$\rightarrow$	# Key-Chains that a Key is in
Two Blocks share ($\lambda = 1$) Objects	$\rightarrow$	Two Key-Chains share (χ) Keys

used as a key-chain. That provides $b \geq N$ key-chains each having $K = k = n+1$ keys. Symmetric Design guarantees that any pair of blocks has λ objects in common, meaning that any pair of key-chains, or equivalently sensor nodes, has $\chi = \lambda$ keys in common.

Construction: There are several methods to construct *Symmetric Designs* of the form $(n^2 + n + 1, n + 1, 1)$. In this project, we use a *complete set* of $(n - 1)$ *Mutually Orthogonal Latin Squares (MOLS).* A *Latin Square* on n symbols is an $n \times n$ array such that each of the n symbols occurs exactly once in each row and each column. The number n is called the *order of the square.* If $A = (a_{ij})$ and $B = (b_{ij})$ are any two $n \times n$ arrays, the *join* of A and B is a $n \times n$ array whose $(i, j)^{th}$ element is the pair (a_{ij}, b_{ij}). The Latin Squares A and B of order n are *Orthogonal* if all entries of the join of A and B are distinct. Latin Squares $A_1, A_2, ..., A_r$ are *Mutually Orthogonal* ($MOLS$) if they are orthogonal in pairs. For prime power n, a set of $(n - 1)$ MOLS of order n is called a *Complete Set* [8, 1]. A complete set of $(n - 1)$ MOLS can be used to construct *Affine Plane* of order n which is an $(n^2, n, 1)$ design. *Affine Plane* of order n can be converted to *Projective Plane* of order n which is a $(n^2+n+1, n+1, 1)$ *Symmetric Design.* The construction algorithm can be summarized as follows:

1. Given a network size of N, find a prime power n where $n^2 + n + 1 \geq N$,
2. Generate a complete set of $(n - 1)$ MOLS of order n [1, Theorem 5.1.1],
3. Construct the Affine Plane of order n from the MOLS [1, Theorem 1.3.5],
4. Construct the Projective Plane of order n from the Affine Plane [1, Theorem 1.2.5].

Analysis: Symmetric Design has a very nice property that, any pair of blocks shares exactly one object. Probability of key share between any pair of nodes is $P_{SYM} = 1$, so that *Average Key Path Length* is 1.

Symmetric Design of the form $(n^2 + n + 1, n + 1, 1)$ is not a scalable solution itself. Given a fixed key-chain size $k = n + 1$, it can support network sizes of N where $N \leq n^2 + n + 1$. For networks smaller than $n^2 + n + 1$, simply some of blocks may not be used still preserving key sharing probability $P_{SYM} = 1$. For the networks where $N > n^2 + n + 1$, key-chain size must be increased, that is, n must be increased to next prime power. Due to the memory limitations in a

Table 2. The $GQ(s,t)$ parameters

GQ(s,t)	s	t	b	v
$GQ(q,q)$	q	q	q^3+q^2+q+1	q^3+q^2+q+1
$GQ(q,q^2)$	q	q^2	$q^5+q^3+q^2+1$	q^4+q^3+q+1
$GQ(q^2,q^3)$	q^2	q^3	$q^8+q^5+q^3+1$	$q^7+q^5+q^2+1$

Table 3. Mapping from GQ to Key Distribution

Generalized Quadrangle $GQ(s,t)$		Key Distribution
Point Set (P)	$\rightarrow$	Key-Pool (P)
Point Set Size ($\|S\| = v = (s+1)(st+1)$)	$\rightarrow$	Key-Pool Size ($\|P\|$)
Line Set (B)	$\rightarrow$	Key-Chains
# Lines ($\|B\| = b = (t+1)(st+1)$)	$\rightarrow$	# Key-Chains (N)
# Lines ($\|B\| = b = (t+1)(st+1)$)	$\rightarrow$	# Sensor Nodes (N)
# Points on a Line ($s+1$)	$\rightarrow$	# Keys in a Key-Chain (K)
# Lines that a Point is incident ($t+1$)	$\rightarrow$	# Key-Chains that a Key is in
Two Lines share (≤ 1) points	$\rightarrow$	Two Key-Chains share (χ) Keys

sensor node, this may not be a good solution. Moreover, such an increase in n may produce designs which can support much bigger networks than required. In probabilistic key distribution schemes, it is always possible to increase size of key-pool for a fixed key-chain size to increase the number of key-chains. But, such an approach sacrifices key share probability and requires better connectivity at underlying physical network. It is possible to merge deterministic and probabilistic designs to inherit advantages of both. Later in Section-4, we propose *Hybrid* of Symmetric and Probabilistic Designs to cope with scalability problems. Basically, we use n^2+n+1 blocks of the Symmetric Design and select uniformly at random remaining $N-(n^2+n+1)$ blocks among the $(k=n+1)$-subsets of the Complementary Symmetric Design.

3.2 Mapping from Generalized Quadrangles to Key Distribution

In this work, we are interested in three known $GQ(s,t)$: $GQ(q,q)$, $GQ(q,q^2)$ and $GQ(q^2,q^3)$. Table-2 gives details about their parameters.

Mapping: Consider a sensor network of N nodes where each node requires a *key-chain* having K keys coming from a *key-pool* P. Assume also that, not all pairs of neighboring nodes need to share a key directly, they can communicate through a secure path on which every pair of neighboring nodes shares a key. GQ can be used to generate key-chains for such networks. Namely, points in GQ can be considered as the keys and lines as the key-chains. Mapping between GQ and Key Distribution is given in Table-3.

In GQ, there are $(t+1)$ lines passing through a point, and a line has $(s+1)$ points. That means, a line shares a point with exactly $t(s+1)$ other lines.

Table 4. Projective Space equations

GQ	PG	Points	Canonical Equation for PG
$GQ(q,q)$	$PG(4,q)$	(x_0,x_1,x_2,x_3,x_4)	$x_0^2+x_1x_2+x_3x_4=0$
$GQ(q,q^2)$	$PG(5,q)$	$(x_0,x_1,x_2,x_3,x_4,x_5)$	$f(x_0,x_1)+x_2x_3+x_4x_5=0$
$GQ(q^2,q^3)$	$PG(4,q^2)$	(x_0,x_1,x_2,x_3,x_4)	$x_0^{q+1}+x_1^{q+1}+x_2^{q+1}+x_3^{q+1}+x_4^{q+1}=0$

Moreover, if two lines, say lines A and B, do not share a point, then for each point pt_A on line A, there is a point pt_B on line B such that there exist a line C passing through both points pt_A and pt_B. That means, if two lines A and B do not share a point, there are $(s+1)$ distinct lines which share a point with both lines A and B. In terms of Key Distribution, it means that, a block shares a key with $t(s+1)$ other blocks. Additionally, if two blocks do not share a key, there are $(s+1)$ other blocks sharing a key with both.

Construction: The three $GQ(s,t)$'s used in this work are incidence relations between points and lines in a *Projective Space* $PG(d,q)$ and $PG(d,q^2)$ with dimension d. Points of the space are vectors with $(d+1)$ elements of the form $(x_0,x_1,x_2,\ldots,x_d)$ where $x_i<q$ for $PG(d,q)$ and $x_i<q^2$ for $PG(d,q^2)$. They hold the projective plane equations given in Table-4.

We use irreducible binary quadratic $f(x_0,x_1)=dx_0^2+x_0x_1+x_1^2$ for $GQ(q,q^2)$ as given in Table-4. Our construction algorithm can be summarized as follows:

1. Given network size of N, find a prime power q where:
   $b=q^3+q^2+q+1 \geq N$ for $GQ(q,q)$.
   $b=q^5+q^3+q^2+1 \geq N$ for $GQ(q,q^2)$.
   $b=q^8+q^5+q^3+1 \geq N$ for $GQ(q^2,q^3)$.
2. Find all points in Projective Space $PG(4,q)$ for $GQ(q,q)$, $PG(5,q)$ for $GQ(q,q^2)$ and $PG(4,q^2)$ for $GQ(q^2,q^3)$. That is, find all points holding given canonical equation.
3. Construct bilinear groups of size $s+1$ from v points, that is, find $s+1$ points which are on the same line. Note that each point is incident to $t+1$ lines.

Analysis: In a $GQ(s,t)$, there are $b=(t+1)(st+1)$ lines and a line intersects with $t(s+1)$ other lines. Thus, in a design generated from a GQ, a block shares an object with $t(s+1)$ other blocks. Probability P_{GQ} that two blocks shares at least one object, or equivalently, probability P_{GQ} that a pair of nodes share at least one key is:

$$P_{GQ} = \frac{t(s+1)}{b} = \frac{t(s+1)}{(t+1)(st+1)} .$$

Table-5 lists key share probabilities for the three GQ.

Probabilistic key distribution is the simplest and most scalable solution when compared to GQ and Symmetric Designs. Next, in Section-4, we propose Hybrid Symmetric and GQ Designs which provide solutions as scalable as probabilistic key distribution schemes, yet taking advantages of underlying GQ and Symmetric Designs.

Table 5. Pairwise Key Sharing Probabilities

GQ	Pairwise Key Sharing Probability
$GQ(q,q)$	$P_{QQ} = \frac{q^2+q}{q^3+q^2+q+1}$
$GQ(q,q^2)$	$P_{QQ^2} = \frac{q^3+q^2}{q^5+q^3+q^2+1}$
$GQ(q^2,q^3)$	$P_{Q^2Q^3} = \frac{q^5+q^3}{q^8+q^5+q^3+1}$

Table 6. Parameters k, r, v, b for Symmetric, GQ and their Complementary Designs

Design	k	r	b	v
Symmetric	$n+1$	$n+1$	n^2+n+1	n^2+n+1
Complementary Symmetric	n^2	n^2	n^2+n+1	n^2+n+1
$GQ(n,n)$	$n+1$	$n+1$	n^3+n^2+n+1	n^3+n^2+n+1
Complementary $GQ(n,n)$	n^3+n^2	n^3+n^2	n^3+n^2+n+1	n^3+n^2+n+1
$GQ(n,n^2)$	$n+1$	n^2+1	$n^5+n^3+n^2+1$	n^4+n^3+n+1
Complementary $GQ(n,n^2)$	n^4+n^3	n^5+n^3	$n^5+n^3+n^2+1$	n^4+n^3+n+1
$GQ(n^2,n^3)$	n^2+1	n^3+1	$n^8+n^5+n^3+1$	$n^7+n^5+n^2+1$
Complementary $GQ(n^2,n^3)$	n^7+n^5	n^8+n^5	$n^8+n^5+n^3+1$	$n^7+n^5+n^2+1$

4 Hybrid Designs for Scalable Key Distributions

The main drawback of the combinatorial approach comes from the difficulty of their construction. Given a desired number of sensor nodes or a desired number of keys in the pool, we may not be able to construct a combinatorial design for the target parameters.

In this work, we present a novel approach called *Hybrid Design* which combines deterministic core and probabilistic extensions. We will consider two Hybrid Designs: *Hybrid Symmetric Design* and *Hybrid GQ Design*. By using Symmetric or GQ Design and its complement, we preserve nice properties of combinatorial design yet take advantages of flexibility and scalability of probabilistic approaches to support any network sizes.

4.1 Mapping

Consider a sensor network where there are N nodes, therefore N key-chains are required. Due to memory limitations, key-chains can have at most K keys coming from key-pool P. We can employ Hybrid Design for the cases where there is no known combinatorial design technique to generate design with N nodes for the given key-chain size K. Basically, Hybrid Design finds largest prime power n such that $k \leq K$ and generates N blocks of size k where objects come from object set S of size $|S| = v$. The b of N blocks are generated by base Symmetric or GQ Design and $N - b$ blocks are randomly selected among k-subsets of the Complementary Design blocks. We define mappings as in Table-7.

Table 7. Mapping from Hybrid Design to Key Distribution

Hybrid Symmetric Design		Key Distribution
Object Set (S)	$\rightarrow$	Key-Pool (P)
Object Set Size ($\|S\| = v$)	$\rightarrow$	Key-Pool Size ($\|P\|$)
Blocks of base design and selected (k)-subsets from Complementary Design	$\rightarrow$	Key-Chains
# blocks from base design (b) + # selected (k)-subsets ($N - b$)	$\rightarrow$	# Key-Chains (N)
# blocks from base design (b) + # selected (k)-subsets ($N - b$)	$\rightarrow$	# Sensor Nodes (N)
# Objects in a Block ($k \leq K$)	$\rightarrow$	# in a Key-Chain (K)
Two Blocks share zero or more Objects	$\rightarrow$	Two Key-Chains share (χ) Keys

4.2 Construction

For a given key-chain size K and network size N, Hybrid Design first generates the Base Symmetric or GQ Design with largest possible prime power n where $k \leq K$. Base Symmetric or GQ Design has b blocks of size k. Table-6 lists the relations between block size k and number of blocks b for the prime power n. Next step is to generate Complementary Design where there are b blocks of size $v - k$. Table-6 lists the parameters of the Complementary Designs. Due to the fact that $v - k > k$ for Symmetric and GQ designs, blocks of the Complementary Design can't be used as the key-chains, but their subsets can. To scale the base design up to given network size, Hybrid Design randomly selects remaining $N - b$ blocks uniformly at random among k-subsets of the Complementary Design blocks. Selected k-subsets along with the blocks of the base design form the Hybrid Design blocks. Algorithm can be summarized as follows:

1. Given N sensor nodes where each can store key-chain of size K, find largest possible prime power n such that $k \leq K$ for k values given in Table-6.
2. Generate base design (Symmetric or GQ):
   - Generate object pool $P = \{a_1, a_2, ..., a_v\}$ of size v,
   - Generate blocks $B = \{B_1, B_2, \ldots, B_b\}$ where $|B_i| = k$ for $1 \leq i \leq b$ and $B_i \subset P$.
3. Generate Complementary Design from the base design:
   - Generate blocks $\overline{B} = \{\overline{B_1}, \overline{B_2}, \ldots, \overline{B_b}\}$ where $\overline{B_i} = P - B_i$ and $|\overline{B_i}| = v - k$ for $1 \leq i \leq b$.
4. Generate $N - b$ hybrid blocks H=$\{H_1, H_2, \ldots, H_{N-b}\}$ of size $|H_i| = k$ ($1 \leq i \leq N - b$) from the Complementary Design $\overline{B} = \{\overline{B_1}, \overline{B_2}, \ldots, \overline{B_b}\}$. Use variable s_i to hold index of the block in $\overline{B}$ from which block H_i is obtained:
   - Consider all k-subsets of all blocks in $\overline{B}$,
   - Randomly select $N - b$ distinct k-subsets to generate the set H,
   - For each selected k-subset H_i ($1 \leq i \leq N - b$), find the block $\overline{B_j} \in \overline{B}$ ($1 \leq j \leq b$) from which block H_i is obtained. Set $s_i = j$.
5. Blocks of the Hybrid Design are $B \cup H$.

Example 1: Assume that we would like to generate key-chains for a network with $N = 10$ nodes. Assume also that nodes have very limited memories, so that they can store at most $K = 3$ keys in their key-chains. Hybrid Symmetric Design can be used to generate design for this network. Symmetric Design $(v, k, \lambda) = (7, 3, 1)$ can be used as the base design to generate $b = 7$ blocks out of $v = 7$ objects where block size is $k = 3$. Blocks of Symmetric Design form the set B={{1,2,3}, {1,4,5}, {1,6,7}, {2,4,6}, {2,5,7}, {3,4,7}, {3,5,6}}. Remaining $N - b = 3$ blocks are selected uniformly at random among the 3-subsets of the Complementary Symmetric Design $\overline{B}$ = {{4,5,6,7}, {2,3,6,7}, {2,3,4,5}, {1,3,5,7}, {1,3,4,6}, {1,2,5,6}, {1,2,4,7}}. Assume that selected blocks are {4,5,6}, {2,3,6} and {1,5,7} which are the 3-subsets of the sets {4,5,6,7}, {2,3,6,7} and {1,3,5,7} respectively. These blocks (3-subsets) form the set H={{4,5,6}, {2,3,6}, {1,5,7}}. The blocks of the Hybrid Symmetric Design is then $B \cup H$= {{1,2,3}, {1,4,5}, {1,6,7}, {2,4,6}, {2,5,7}, {3,4,7}, {3,5,6}, {4,5,6}, {2,3,6}, {1,5,7}}.

4.3 Analysis

In this section, we analyze some important properties of Hybrid Symmetric and Hybrid GQ Designs. We will look for some useful properties coming from underlying combinatorial design. Based on these properties, we will analyze object share probabilities between any pair of blocks in Hybrid Design $B \cup H$, where B is the set of blocks of the base (Symmetric or GQ) design and H is the set of blocks which are uniformly at random selected among k-subsets of the complement design blocks $\overline{B}$ (variable s_i holds index of the block in $\overline{B}$ from which block $H_i \in H$ is obtained).

Hybrid Symmetric Design

Property 1. Given Hybrid Design $B \cup H$, $\forall\ \beta \in B$ and $\theta \in H$, $\exists b \in \beta | b \notin \theta$. □

Proof. For the proofs of this property and the others please refer to [4].

Property 1 doesn't hold among the blocks in H. To see that, consider two such distinct blocks $H_i \in H$ and $H_j \in H$ where $s_i \neq s_j$. Complementary Design of a Symmetric Design has the property that, any pair of blocks has $n^2 - n$ objects in common. For $n > 2$, when $(n^2 - n) > (n + 1)$, it can be the case that randomly selected blocks (k-subsets) H_i and H_j are equivalent.

Property 2. Given key chain size $k = n + 1$, Hybrid Symmetric Design can support network sizes up to:
$\binom{v}{k} = \binom{n^2+n+1}{n+1}$. □

This is the maximum network size that simple probabilistic key pre-distribution scheme can support for key-chain size $k = n+1$ and key-pool size $v = n^2+n+1$. Probabilistic scheme can go beyond this limit by simply increasing the key-pool size v for a fixed key-chain size k. To provide the same scalability, we employ

Hybrid GQ Designs which is analyzed in the next section. For fixed key chain size $k = n + 1$, $GQ(n, n^2)$ will be able to generate designs for networks up to:
$\binom{v}{k} = \binom{n^4+n^3+n+1}{n+1}$.
This is the upper limit of our deterministic algorithms. Numerically, for key chain size of 4, our Hybrid $GQ(n, n^2)$ Design supports network sizes up to $6,210,820$. It supports (2.54×10^{14}) nodes for $k = 6$, (8.08×10^{22}) nodes for $k = 8$, (1.18×10^{32}) nodes for $k = 10$, (5.78×10^{41}) nodes for $k = 12$ and so on.

Theorem 1. *Probability P_{HSYM} that any pair of blocks shares a key in Hybrid Symmetric Design is:*
$P_{HSYM} \leq \frac{b(b-1)}{N(N-1)} \times 1 + \frac{2b(N-b)}{N(N-1)} \times \frac{n^2+2}{n^2+n+1} + \frac{(N-b)(N-2b)}{bN(N-1)} \times P_H + \frac{(b-1)(N-b)^2}{bN(N-1)} \times 1$.

$P_{HSYM} \geq \frac{b(b-1)}{N(N-1)} \times 1 + \frac{2b(N-b)}{N(N-1)} \times \frac{\frac{1}{2}n^2+\frac{3}{2}n+1}{n^2+n+1} + \frac{(N-b)(N-2b)}{bN(N-1)} \times P_H + \frac{(b-1)(N-b)^2}{bN(N-1)} \times 1$.

Where $P_H = \left[1 - \frac{\binom{n^2-n-1}{n+1}}{\binom{n^2}{n+1}}\right]$. □

Hybrid GQ Designs

Property 3. Given key chain size $k = n + 1$, Hybrid GQ Design can support network sizes up to:
$\binom{v}{s+1} = \binom{(s+1)(st+1)}{s+1}$. □

Theorem 2. *Probability P_{HGQ} that any pair of blocks shares a key in Hybrid GQ Design is:*
$P_{HGQ} \leq \frac{b(b-1)}{N(N-1)} \times P_{GQ} + \frac{2b(N-b)}{N(N-1)} \times \frac{(st-s+t+2)}{(t+1)(st+1)} + \frac{(N-b)(N-2b)}{bN(N-1)} \times P_H + \frac{(b-1)(N-b)^2}{bN(N-1)}$.

$P_{HGQ} \geq \frac{b(b-1)}{N(N-1)} \times P_{GQ} + \frac{2b(N-b)}{N(N-1)} \times \frac{(s+1)(t-s/2+1)}{(t+1)(st+1)} + \frac{(N-b)(N-2b)}{bN(N-1)} \times P_H + \frac{(b-1)(N-b)^2}{bN(N-1)}$.

Where P_{GQ} is given in Table-5 and $P_H = \left[1 - \frac{\binom{(s+1)(st-1)}{s+1}}{\binom{st(s+1)}{s+1}}\right]$. □

5 Computational Results

We have implemented *Random Key Pre-distribution Scheme* by Eschenauer *et al.* [14], *Symmetric Design*, $GQ(q, q)$, $GQ(q, q^2)$, *Hybrid Symmetric Design*, and compared them with each other. In random key pre-distribution scheme, we initially generate a large pool of P keys and their identities. For each sensor, we uniformly at random draw k keys from the key-pool P without replacement. These k keys and key identities form the *key-chain* for a sensor node.

Basically, for a network of size N, we generate N key-chains and assign them to N sensor nodes. Then, we uniformly randomly distribute N nodes in to a 1×1 unit grid. Every wireless sensor has a coverage of radius r where $r = d\,(ln\,N)/N$, every node within this coverage area is assumed to be a neighbor. Note that,

Table 8. Symmetric Design vs Random Key Pre-distribution

Pool Size (P)	Key Chain Size(k)	Number Sensor Nodes	Random Prob.	Symmetric Prob.	Random Avg. Key Path	Symmetric Avg. Key Path	Avg. Node Degree
100807	318	100807	0.634	1.0	–	1.0	–
10303	102	10303	0.639	1.0	1.35	1.0	56
5113	72	5113	0.642	1.0	1.35	1.0	51
2863	54	2863	0.645	1.0	1.35	1.0	47
1407	38	1407	0.651	1.0	1.34	1.0	42
553	24	553	0.663	1.0	1.33	1.0	35

parameter d can be used to play with radius r and therefore average degree of the network.

After the deployment, two neighboring nodes compare the keys in their key-chains by using the key id's. If they have a key in common, it is used to secure the communication. If there is no key in common, they try to find a shortest possible path where each pair of nodes on the path shares a key. Length of this path is called *Key Path Length* where *Key Path Length* for two nodes directly sharing a key is 1. *Average Key Path Length* is one of the metrics that we use to compare random key pre-distribution scheme with our Combinatorial and Hybrid Design schemes.

Probability p that two key-chains share at least one key is another metric we use in comparison. For random key pre-distribution scheme, for a given key-pool size P and key-chain size k, Eschenauer *et al.* [14] approximate probability p as: $P_{RAND} = \left[1 - \frac{(1-\frac{k}{P})^{2(P-k+1/2)}}{(1-\frac{2k}{P})^{(P-2k+1/2)}}\right]$.

In Symmetric Design, $P_{SYM} = 1$ since any pair of key-chains shares exactly one key. In $GQ(s,t)$, probability of key share P_{QQ} for $GQ(q,q)$, P_{QQ^2} for $GQ(q,q^2)$ and $P_{Q^2Q^3}$ for $GQ(q^2,q^3)$ is given in Table-5.

Probability of key share P_{HSYM} is given in analysis section of the Hybrid Symmetric Design. Similarly, probability of key share P_{HGQ} for Hybrid GQ Design is given in analysis section of the Hybrid GQ Designs.

Tables 8, 9 and 10 summarize the computational results: (i) analytical solution for probability p that two key-chains share at least one key, and (ii) simulation results for *Average Key Path Length*.

Symmetric Design is compared with Random Key Pre-distribution scheme in Table-8. For the same network size, key-chain size and pool-size, Symmetric Design provides better probability of key share between any two key-chains. Simulation results for average key path length supports this advantage. In Random Key Pre-distribution scheme, a pair of nodes requires to go through a path of 1.35 hops on average to share a key and communicate securely. This path length is 1 for Symmetric Design.

$GQ(q,q)$ is compared with Random Key Pre-distribution scheme in Table-9. $GQ(q,q)$ decreases key-chain size, causing a small decrease in key sharing probability. Analytical solution shows that random key pre-distribution scheme

Table 9. Generalized Quadrangle $GQ(q, q)$ vs Random Key Pre-distribution

Pool Size (P)	Key Chain Size(k)	Number Sensor Nodes	Random Prob.	$GQ(q, q)$ Prob.	Random Avg. Key Path	$GQ(q, q)$ Avg. Key Path	Avg. Node Degree
7240	20	7240	0.053	0.052	2.68	2.69	205
5220	18	5220	0.060	0.058	2.89	2.88	148
2380	14	2380	0.079	0.076	3.17	3.18	88
1464	12	1464	0.094	0.090	2.73	2.71	81
400	8	400	0.150	0.140	3.61	3.49	32
156	6	156	0.212	0.192	2.82	2.53	25

Table 10. Hybrid Symmetric Design vs Random Key Pre-distribution

Pool Size (P)	Key Chain Size(k)	Number Sensor Nodes	Random Prob.	Hybrid Sym. Prob.	Random Avg. Key Path	Hybrid Sym. Avg. Key Path	Avg. Node Degree
10303	102	10500	0.632	0.99	1.36	1.01	56
5113	72	5250	0.632	0.99	1.35	1.01	51
2863	54	3000	0.628	0.98	1.35	1.03	47
1407	38	1500	0.627	0.97	1.34	1.04	42
553	24	750	0.547	0.89	1.33	1.15	37
183	14	250	0.563	0.89	1.31	1.14	29

provides slightly better probability of key share between key-chains, but $GQ(q, q)$ is still competitive to random key pre-distribution scheme. When two key-chains do not share a key, $GQ(q, q)$ guarantees existence of third one which shares a key with both.

Hybrid Symmetric Design is compared with Random Key Pre-distribution Scheme in Table-10. Hybrid Symmetric Design makes use of Symmetric Design, yet taking advantages of the scalability of probabilistic approach. Given target network size N and key chain size k for which there is no known design, computational results shows that Hybrid Symmetric Design shows better performance than Probabilistic Design.

6 Conclusions

In this work we presented novel approaches to the key distribution problem in large scale sensor networks. In contrast with prior work, our approach is combinatorial based on Combinatorial Block Designs. We showed how to map from two classes of combinatorial designs to deterministic key distribution mechanisms. We remarked the scalability issues in the deterministic constructions and proposed hybrid mechanisms. Hybrid constructions combine a deterministic core design with probabilistic extensions to achieve key distributions to any network size.

The analysis and computational comparison to the randomized methods show that the combinatorial approach has clear advantages: (i) it increases the probability of a pair of sensor nodes to share a key, and (ii) decreases the key-path length while provides scalability with hybrid approaches.

References

1. I. Anderson, "Combinatorial Designs: Construction Methods," Ellis Horwood Limited, 1990.
2. R. Blom, "An optimal class of symmetric key generation systems," EUROCRYPT 84, 1985.
3. C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung, "Perfectly-secure key distribution for dynamic conferences," In Advances in Cryptography - CRYPTO'92, 1993.
4. S. A. Camtepe, B. Yener, "Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks," RPI Computer Science Department, Technical Report 04-10, www.cs.rpi.edu/research/tr.html, 2004.
5. D.W. Carman, B.J. Matt and G.H. Cirincione, "Energy-efficient and Low-latency Key Management for Sensor Networks", In Proceedings of 23rd Army Science Conference, 2002.
6. H. Chan, A. Perrig and D. Song, "Random Key Predistribution Schemes for Sensor Networks," In 2003 IEEE Symposium on Research in Security and Privacy, 2003.
7. M. Chen, W. Cui, V. Wen and A. Woo, "Security and Deployment Issues in a Sensor Network," Ninja Project, A Scalable Internet Services Architecture, Berkeley, http://citeseer.nj.nec.com/chen00security.html, 2000.
8. C.J. Colbourn, J.H. Dinitz, "The CRC Handbook of Combinatorial Designs," CRC Press, 1996.
9. W. Du, J. Deng, Y. S. Han, P. Varshney, "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks," In Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS), 2003.
10. W. Du, J. Deng, Y. S. Han, S. Chen, P. K. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," INFOCOM, 2004.
11. J. Deng, R. Han and S. Mishra, "Enhancing Base Station Security in Wireless Sensor Networks," Technical Report CU-CS-951-03, Department of Computer Science, University of Colorado, 2003.
12. J. Deng, R. Han, and S. Mishra, "A Performance Evaluation of Intrusion-Tolerant Routing in Wireless Sensor Networks," 2nd International Workshop on Information Processing in Sensor Networks (IPSN '03), 2003.
13. P. Dembowski, "Finite Geometries," Springer Verlag, 1968.
14. L. Eschenauer, V. D. Gligor, "A key-management scheme for distributed sensor networks", Proceedings of the 9th ACM conference on Computer and communications security, 2002.
15. M. Hall, "Combinatorial Theory," Blaisdell Publishing Company, 1967.
16. J.W.P. Hirschfeld, "Projective Geometries Over Finite Fields," Clarendon Press Oxford, 1979.
17. D. Liu and P. Ning, "Efficient Distribution of Key Chain Commitments for Broadcast Authentication in Distributed Sensor Networks", The 10th Annual Network and Distributed System Security Symposium, February 2003

18. D. Liu, P. Ning, K. Sun, "Efficient self-healing group key distribution with revocation capability," Proceedings of the 10th ACM conference on Computer and communication security, 2003.
19. D. Liu, P. Ning, "Establishing pairwise keys in distributed sensor networks," Proceedings of the 10th ACM conference on Computer and communication security, 2003.
20. R. Merkle, "Secure Communication over insecure channels," Communications of the ACM, 1978.
21. S. E. Payne, J. A. Thas, "Finite Generalized Quadrangles," Research Notes in Mathematics, Pitman Advanced Publishing Program, 1984.
22. D. Pedoe, "An introduction to Projective Geometry," Oxford, 1963.
23. A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," Wireless Networks Journal (WINE), 2002.
24. S. Slijepcevic, M. Potkonjak, V. Tsiatsis, S. Zimbeck, M. B. Srivastava, "On communication Security in Wireless Ad-Hoc Sensor Network," Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02), 2002.
25. F. Stajano, R. Anderson, "The resurrecting duckling: security issues for ad-hoc wireless networks," AT&T software symposium, 1999.
26. D. R. Stinson, S. A. Vanstone, "A combinatorial approach to threshold schemes,", Advances in Cryptology - CRYPTO '87, 1987.
27. D. R. Stinson, "A construction for authentication / secrecy codes from certain combinatorial designs," Advances in Cryptology - CRYPTO '87, 1987.
28. D. R. Stinson, "Combinatorial characterizations of authentication codes," Advances in Cryptology - CRYPTO '91, 1991.
29. Y. Song, A. Wool, B. Yener, "Combinatorial Design of Multi-ring Networks with Combined Routing and Flow Control," in Computer Networks Vol.3 No: 3, pp 247-267, 2003.
30. J. Undercoffer, S. Avancha, A. Joshi, and J. Pinkston, "Security for Sensor Networks," CADIP Research Symposium, 2002.
31. W.D. Wallis, "Combinatorial Desing," Marcel Dekker Inc., 1988.
32. B. Yener, Y. Ofek, M. Yung, "Combinatorial Design of Congestion Free Networks," In IEEE/ACM Transactions on Networking, Vol. 5, No. 6, pages: 989-1000, December 1997.
33. S. Zhu, S. Xu, S. Setia, S. Jajodia, "Establishing Pairwise Keys for Secure Communication in Ad Hoc Networks: A Probabilistic Approach," 11th IEEE International Conference on Network Protocols (ICNP'03), 2003.
34. S. Zhu, S. Setia, S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," Proceedings of the 10th ACM conference on Computer and communication security, 2003.