A New Scalable Key Pre-distribution Scheme for WSN

Walid Bechkit, Yacine Challal, Abdelmadjid Bouabdallah

A New Scalable Key Pre-distribution Scheme for WSN

Walid Bechkit *, Yacine Challal * and Abdelmadjid Bouabdallah *
* Compiegne University of Technology , HeuDiaSyc laboratory, UMR CNRS 6599, Compiegne, France
Email: {wbechkit,ychallal,bouabdal}@hds.utc.fr

***Abstract*—Given the sensitivity of the potential applications of wireless sensor networks, security emerges as a challenging issue in these networks. Because of the resource limitations, symmetric key establishment is one favorite paradigm for securing WSN. One of the main concerns when designing a key management scheme for WSN is the network scalability. Indeed, the protocol should support a large number of nodes to enable a large scale deployment of the network. In this paper, we propose a new highly scalable key establishment scheme for WSN. For that purpose, we make use, for the first time, of the unital design theory. We show that the basic mapping from unitals to pairwise key establishment allows to achieve an extremely high network scalability while degrading, however, the key sharing probability. We propose then an enhanced unital-based pre-distribution approach which provides high network scalability and good key sharing probability. We conduct analytic calculation and simulations to compare our solutions to existing ones regarding different criteria. The obtained results show that our approach enhances considerably the network scalability while providing good overall performances. We show also that our solutions reduce significantly the storage overhead at equal network size compared to existing solutions.**

***Index Terms*—Wireless sensor networks, security, key management, network scalability, resource optimization.**

I. Introduction

Nowadays, wireless sensor networks (WSN) are increasingly used in numerous fields such as military, medical and industrial sectors; they are more and more involved in several sensitive applications which require sophisticated security services [1]. Due to the resource limitations, existing security solutions for conventional networks could not be used in WSN. So, the security issues became then one of the main challenges for the resource constrained environment of WSN. Key management is a corner stone service for many security services such as confidentiality and authentication which are required to secure communications in WSN. The establishment of secure links between nodes is then a challenging problem in WSN. The public key based solutions, which provide efficient key management services in conventional networks, are unsuitable for WSN because of resource limitations. Some public key schemes have been implemented on real sensors [2][3][4], however most researchers believe that these techniques are still too heavyweight over actual sensors' technology because they induce an important communication and computation overhead [5]. Symmetric key establishment is then one of the most suitable paradigms for securing exchanges in WSN. Because of the lack of infrastructure in WSN, we have usually no trusted third party which can attribute pairwise secret keys to neighboring nodes, that is why most existing solutions are based on key pre-distribution.

In this paper, we are interested in particular in the scalability of symmetric key pre-distribution schemes. Existing research works either allow to support a low number a nodes or degrade the other network performances including resiliency, connectivity and storage overhead when the number of nodes is important. In contrast to these solutions, our goal is to enhance the scalability of WSN key management schemes without degrading significantly the other network performances. To achieve this goal, we propose to use, for the first time, the unital design to construct and pre-distribute key rings. First, we explain the unital design and we propose a basic mapping from unitals to key pre-distribution for WSN. We show through analytic calculations that the resulting basic scheme allows to achieve an extremely high network scalability while degrading, however, the key sharing probability. For this, we propose an enhanced unital-based construction in order to maintain a good key sharing probability while enhancing the network scalability. We carried out analytic calculations and simulations to compare the efficiency of the enhanced proposed approach against basic schemes with respect to important performance criteria: storage overhead, network scalability, session key sharing probability and average secure path length. The obtained results show that at equal key ring size, our approach enhances considerably the network scalability while providing good overall performances. Moreover, we show that given a network size, our solutions reduce significantly the key ring size and then the storage overhead compared to existing solutions.

The remainder of this paper is organized as follows: We define in section 2 the metrics used to evaluate and compare key pre-distribution schemes and we summarize the used symbols. Section 3 presents some related works. We give in section 4 a background on unital design while we present, in section 5, the basic mapping to key pre-distribution and analyze the performances of the resulting scheme. In section 6, we present the enhanced unital-based construction. In section 7, we evaluate the performances of the enhanced scheme and compare it to the existing ones with respect to various performance criteria; we provide and discuss theoretical and simulation results. Finally, section 8 ends up this paper with some conclusions and future works.

II. Evaluation metrics and used symbols

In this work, we consider mainly four metrics to compare performances of our solutions against existing ones:

i) Network scalability : represents the maximum number of generated key rings which corresponds to the maximum number of supported nodes. A large scale secure deployment of sensor networks relies strongly on this performance metric.

ii) Storage overhead : measures the memory required to store keys in each node. Because of their small size, sensor nodes are very constrained in term of memory resource and this metric is challenging. We focus, in this work, on the memory required to store keys and we omit the memory required to store the key identifiers when they are necessary. The key identifier size can be computed as the 2-logarithm of the maximum number of keys used by the protocol which is negligible compared to the key size.

iii) Probability of sharing a session key: computed as the probability that a given pair of neighboring nodes are able to establish a direct secure link through one or more common shared pre-deployed keys. This metric can also be seen as the fraction of secured direct links among possible links in the network.

iv) Average secure path length : when two neighboring nodes have no common keys, they should establish a secure path composed of successive secure links. This metric measures then the average length in hop count of these secure paths.

We summarize in table I the main symbols that we use in the remainder of this paper:

TABLE I
SUMMARY OF NOTATIONS

S	The global key pool
$\|S\|$	The size of the global key pool
KR_i	The key ring of node i
$\|KR_i\|$	The size of the node i key ring
n	The network size (number of nodes)
l	The key size
m	The design order (SBIBD and Unital)
k	Size of a block of a given design
(q, k)	The two parameters of the Ruj et al. trade construction. (k is the block size)
$p(i)$	The probability that two nodes share exactly i keys in their subset of keys
P_c	The probability that two nodes can establish a secure link

III. Related works: key management schemes for WSN

Key management problem in WSN has been extensively studied in the literature and several solutions have been proposed. Many classifications of existing symmetric key management schemes can be found in [6][7][8].

Eschenauer and Gligor proposed in [9] the basic Random Key Pre-distribution scheme denoted by RKP. In this scheme, each node is pre-loaded with a key ring of k keys randomly selected from a large pool S of keys. After the deployment step, each node exchanges with each of its neighbors the list of key identifiers that it maintains in order to identify the common keys. If two neighbors share at least one key, they establish a secure link and compute their session secret key which is one of the common keys. Otherwise, if neighboring nodes do not have common keys, they should determine secure paths which are composed of successive secure links. This basic approach is CPU and energy efficient but it requires a large memory space to store the key ring. Moreover, if the network nodes are progressively corrupted, the attacker may discover a large part or the whole global key pool. Hence, a great number of links will be compromised.

Chan et al. proposed in [10] the Q-composite scheme which enhances the resilience of RKP. In this solution, two neighboring nodes can establish a secure link only if they share at least Q keys. The pairwise session key is calculated as the hash of all shared keys concatenated to each other. This approach enhances the resilience against node capture attacks because the attacker needs more overlap keys to break a secure link. However, this approach degrades the probability of session key sharing neighboring nodes must have at least Q common keys to establish a secure link.

Chan et al. proposed also in [10] a perfect secure pairwise key pre-distribution scheme where they assign to each possible link between two nodes i and j a distinct key $k_{i,j}$. Prior to deployment, each node is pre-loaded with $p * n$ keys, where n is the network size and p is the desired secure coverage probability. Hence, the probability that the key $k_{i,j}$ belongs to the key set of the node i is p. Since we use distinct keys to secure each pairwise link, the resiliency against node capture is perfect and any node that is captured reveals no information about links that are not directly connected to it. The main drawback of this scheme is the non scalability because the number of the stored keys depends linearly on the network size. In addition, this solution does not allow the node post-deployment because existing nodes do not have the new nodes' keys.

Du et al. proposed in [11] an enhanced random scheme assuming the node deployment knowledge. Nodes are organized in regional groups to which is assigned different key pools and each node selects its k keys from the corresponding key pool. The key pools are constructed in such a way that neighboring ones share more keys while pools far away from each other share fewer keys. This approach allows to enhance the probability of sharing common keys because the key pools become smaller. Moreover, the network resiliency is improved since if some nodes of a given region are captured, the attacker could discover only a part of the corresponding group key pool. However, the application of this scheme is restrictive since the deployment knowledge of a WSN is not always possible.

In [12], Liu and Ning proposed a new pool based polynomial pre-distribution scheme for WSN. This approach can be considered as an extension of the basic RKP scheme where nodes are pre-loaded with bivariate polynomials instead of keys. A global pool of symmetric bivariate polynomials is generated off-line and each node is pre-loaded with a subset of polynomials. If two neighboring nodes share a common polynomial, they establish a direct

secure by computing the polynomial value at the neighbor identifier; else, they try to find a multi-hop secure path. This approach allows to compute distinct secret keys, so the resilience against node capture is enhanced. However, it requires more memory to store the polynomials and induces more computational overhead.

In [13], Blom proposed a λ-secure symmetric key generation system in which each node i stores a column i and a row i of size $(\lambda + 1)$ of two matrices G and $(D * G)^T$ respectively where : $D_{(\lambda+1)\times(\lambda+1)}$ is a symmetric matrix, $G_{(\lambda+1)\times n}$ is a public matrix and $(D * G)^T$ is a secret matrix. The matrix of pairwise keys of a group of n nodes is then $K = (D*G)^T G$. Yu and Guan [14] used the Blom's scheme to key pre-distribution in group-based WSN. Nodes are deployed into a grid and each group is assigned a distinct secret matrix. Using deployment knowledge, the potential number of neighboring nodes decreases which requires less memory. The application of this scheme is restrictive if the deployment knowledge is not possible.

Liu et al. proposed in [15] SBK, a self-configuring key establishment scheme for WSN. SBK distinguishes two kinds of nodes: service nodes and worker ones. After the deployment, sensor nodes differentiate their role thanks to a pre-loaded bootsrap program. Service nodes generate a key space using a polynomial-based or the matrix-based model. Then, they distribute the corresponding keying shares to at most λ worker nodes. Authors propose for that to use a computationally asymmetric channel based on Rabins public key cryptosystem while shifting the large amount of computation overhead to the service nodes. This induces a high load on service nodes which are sacrificed. SBK assumes that all nodes are deployed at the same time and that they are coarsely time synchronized to to start the bootstrapping procedure simultaneously. it assumes also that the network is secured and no active attacks can be launched during the bootstrapping procedure. SBK gives good performances including scalability, resilience and connectivity between worker nodes as far as the assumptions are verified.

Deterministic key pre-distribution schemes ensure that each node is able to establish a pair-wise key with all its neighbors. A naive deterministic key pre-distribution scheme can be designed by assigning to each link (i,j) a distinct key $K_{i,j}$ and pre-loading each node with $(n-1)$ pairwise keys in which it is involved where n is the network size.The main drawback of this scheme is the non scalability because the number of the stored keys is equal to the network size which is very restrictive. Choi et al. proposed in [16] an enhanced approach allowing to store only (n+1)/2 keys at each node. For that purpose, authors propose to establish an order relation between node identifiers and propose a hash function based key establishment in order to store only half of the node symmetric keys while computing the other half at each node. This approach allows to reduce the required stored keys to the half, however we believe that this scheme remains non scalable enough. Indeed each node should store (n+1)/2 keys, where n is the network size, which could be extremely costly in large scale WSN.

LEAP [17] makes use of a common transitory key which is preloaded into all nodes prior to deployment of the WSN. The transitory key is used to generate pairwise session keys and is cleared from the memory of nodes by the end of a short time interval after their deployment. LEAP is based on the assumption that a sensor node, after its deployment, is secure during a time T_{min} and cannot be compromised during this period of time. LEAP is then secure as far as this assumption is verified.

In [18], Camtepe and Yener proposed a new deterministic key pre-distribution scheme based on Symmetric Balanced Incomplete Block Design (SBIBD). Given a finite set X of ν objects, a BIBD is defined to be a set of k-distinct element subsets of X, called blocks, constructed is such a way that each object occurs in exactly r different blocks and every pair of distinct objects appears together in λ blocks. The number of resulting blocks is b. A BIBD (ν, b, r, k, λ) has two properties: (i) $\lambda(\nu - 1) = r(k - 1)$ and (ii) $bk = \nu r$. A BIBD is called symmetric (SBIBD) when $b = \nu$ and in consequence $r = k$. A SBIBD has the properties: *i)* every block contains k elements; *ii)* each element occurs in exactly k blocks; *iii)* each pair of elements occurs in λ blocks and *iv)* each pair of blocks intersects in λ elements. Camtepe and Yener introduce in [18] a mapping from the SBIBD to the pool based key distribution. To each object is associated a distinct key, to the global set of objects is associated the key pool and to each block is associated the node key ring. We can then generate from a global key pool of $|S|$ keys, $|S|$ key rings of $m + 1$ keys in such a way that each two key rings shares exactly λ keys. The main strength of the Camtepe scheme is the total secure connectivity. Indeed, the use of SBIBD ensures that each two key rings share exactly one common key. However, the SBIBD scheme does not scale to very large networks. Indeed, using key rings of $m+1$ keys we can generate only $m^2 + m + 1$ key rings.

In [19], authors propose a new key management scheme for grid group WSN. Intra-region secure communications are guaranteed thanks to a SBIBD key pre-distribution while inter-region communications are ensured by special nodes called agents. Furthermore, authors propose to enhance the Camtepe scheme in order to avoid key identifier exchanges. For that, they index all nodes and keys and propose a mapping between node indexes and key indexes.

In [20], Ruj et al. propose a new trade-based key pre-distribution scheme for WSN. Given a finite set X of v elements, a *trade* $t - (v, k)$ is defined to be two sets of blocks T_1 and T_2 such as each set contains m blocks of k elements of X. The trade have the properties that: $T_1 \cap T_2 = \varnothing$ and that each set of t elements from X occurs in precisely the same number of blocks of T_1 as those of T_2. In a *Steiner Trade* no set of t elements from X is repeated more than once in any of T_1 or T_2. A $2 - (v, k)$ steiner trade is said to be strong if any two blocks of T_1 and T_2 respectively intersects in at most two elements.

Ruj et al. propose in [20] a new trade construction: Having q a prime power and $(4 \leq k < q)$, they construct T_1 and T_2 while the blocks of T_1 are represented by $t^1_{i,j}$

such that $t^1_{i,j} = \{(x, (xi + j)\ mod\ q) : 0 \leq x < k\}$, where $0 \leq i, j < q$, and the blocks of T_2 are represented by $t^2_{i,j}$ such that $t^2_{i,j} = \left\{(x, (x^2 + xi + j)\ mod\ q) : 0 \leq x < k\right\}$, where $0 \leq i, j < q$. Authors proved that the proposed construction results in a $2-(qk, k)$ strong steiner trade with two sets T_1 and T_2 having q^2 blocks each one while each block has k elements. Authors propose then a mapping to key pre-distribution where they associate to each element a distinct key and to each block of T_1 and T_2 a key ring. The key ring size is then equal to k and the scalability of the scheme is equal to $2q^2$.

After the deployment step, each two nodes can establish a direct secure link if they share exactly two common keys. The pairwise secret key is then computed as the hash of the two common keys. Authors prove that each pair of keys occurs either in exactly two nodes from T_1 and T_2 respectively or none of the nodes. This allows to establish a unique pairwise key for each secured link.

The main strength of the proposed scheme is the establishment of unique secret pairwise keys between connected nodes. However, this does not ensure a perfect network resilience. Indeed, the attacker may construct a part of the global set of keys and then compute pairwise secret keys used to secure external links where the compromised nodes are not involved. Moreover, the proposed scheme provides a low session key sharing probability which does not exceed 0.25 in the best case as we prove later. Another drawback of this solution is the network scalability which reaches only $2q^2 = O(k^2)$ where k is the key ring size.

We focus in this work on the scalability of key management schemes for WSN. Basic schemes giving a perfect network resilience [10] [16] achieve a network scalability of $O(k)$ where k is the key ring size. Design based schemes as the SBIBD [18] and the trade based [20] ones allow to achieve a network scalability of $O(k^2)$. So, large scale networks cannot be supported because the key ring size may be increased which is not suitable due to memory constraints in WSN. In this work we propose new solutions achieving a network scalability of $O(k^4)$ when providing good overall performances. For this purpose, we make use, for the first time, of the unital design in order to pre-distribute keys. We show that the basic use of unital design enhances considerably the scalability of key pre-distribution while decreasing the probability of sharing common keys. In order to achieve a good trade-off between scalability and connectivity, we propose a solution which ensures high network scalability while maintaining a good probability of sharing common keys.

IV. Background: Unital Design

In combinatorics, the design theory deals with the existence and construction of systems of finite sets whose intersections have specified numerical properties. Formally, A t-design (ν, b, r, k, λ) is defined as follows : Given a finite set X of ν points (elements), we construct a family of b subsets of X, called blocks, such that each block has a size k, each point is contained in r blocks and each t points are contained together in exactly λ blocks. For instance, the Symmetric Balanced Incomplete Block Design (SBIBD) presented above is a (ν, b, r, k, λ) design, where $\nu = b = m^2 + m + 1$, $r = k = m + 1$ and $\lambda = 1$.

$$\begin{pmatrix}
1 & . & . & . & . & 1 & 1 & . & . & . & 1 & . \\
. & . & . & . & . & 1 & . & 1 & 1 & 1 & . & . \\
. & . & 1 & 1 & . & . & 1 & 1 & . & . & . & . \\
1 & . & . & 1 & 1 & . & . & . & 1 & . & . & . \\
. & . & . & . & 1 & . & 1 & . & . & 1 & . & 1 \\
. & . & 1 & . & . & . & . & . & 1 & . & 1 & 1 \\
. & 1 & 1 & . & 1 & 1 & . & . & . & . & . & . \\
1 & 1 & . & . & . & . & . & 1 & . & . & . & 1 \\
. & 1 & . & 1 & . & . & . & . & . & 1 & 1 & .
\end{pmatrix}$$

Fig. 1. Example of incidence matrix of a 2-(9,3,1) hermitian unital

A Unital design is a Steiner 2-design which consists of $b = m^2(m^3 + 1)/(m + 1)$ blocks, of a set of $v = m^3 + 1$ points [21]. Each block contains $m + 1$ points and each point is contained in $r = m^2$ blocks. Each pair of points is contained together in exactly one block. We note the Unital as a $2-design(m^3+1, m^2(m^3+1)/(m+1), m^2, m+1, 1)$ or as $(m^3 + 1, m + 1, 1)$ for simplicity sake.

Without loss of generality, we focus in this paper on Hermitian Unitals which exist for all m a prime power. Other construction for m not necessarily a prime power exist in literature [21]. Some Hermitian unital construction approaches were proposed in the literature [22] [23]. We refer in this paper to the construction proposed in [23].

A unital may be represented by its $v \times b$ incidence matrix that we call M. In this matrix, rows represent the points P_i and columns represent blocks B_j. The matrix M is then defined as:

$$M_{ij} = \begin{cases} 1 & \text{if } P_i \in B_j \\ 0 & \text{otherwise} \end{cases}$$

We give in figure 1 an incidence matrix of a 2-(9,3,1) hermitian unital. It consists of 12 blocks of a set of 9 points. Each block contains 3 points and each point occurs in 4 blocks. Each pair of points is contained together in exactly one block.

V. A basic mapping from unitals to key pre-distribution for WSN

At the best of our knowledge, we are the first who propose the use of unital design for pre-distribution in WSN. This scheme may also be generalized to all resource constrained wireless networks where key pre-distribution should be useful. In this section, develop a naive and scalable key pre-distribution scheme based on unital design. We propose a basic mapping in which we associate to each point of the unital a distinct key, to the global set of points the key pool and to each block a node key ring (see table II). We can then generate from a global key pool of $|S| = m^3 + 1$ keys, $n = b = m^2(m^3 + 1)/(m + 1)$ key rings of $k = m + 1$ keys each one.

Before the deployment phase, we generate the unital blocks corresponding to key rings. Each node is then pre-loaded with a distinct key ring as well as the key identifiers.

TABLE II
MAPPING FROM UNITAL DESIGN TO KEY DISTRIBUTION

Unital design	Key distribution
X: Point set	S : Key pool
Blocks	Key rings ($< KR_i >$)
Size of a block ($k = m + 1$)	size of a key ring ($\|KR_i\| = m+1$)
Size of the object set X: $\nu = m^3 + 1$	Size of the key pool S: $\|S\| = m^3 + 1$
Number of generated blocks: $b = m^2(m^2 - m + 1)$	Number of generated key rings (supported nodes) : $n = m^2(m^2 - m + 1)$
Each point belongs to exactly m^2 blocks	each key appears in exactly m^2 key rings

After the deployment step, each two neighboring nodes exchange their key identifiers in order to determine eventual common key. Using this basic approach, each two nodes share at most one common key. Indeed, referring to the unital properties, each pair of keys is contained together in exactly one block which implies that two blocks cannot share more than one point. Hence, if two neighboring nodes share one common key, the latter is used as a pairwise key to secure the link; otherwise, nodes should determine secure paths which are composed of successive secure links.

A. Storage Overhead

When using the proposed naive unital based version matching a unital of order m, each node is pre-loaded with one key ring corresponding to one block from the design. Hence, each node is pre-loaded with $(m+1)$ disjoint keys. The memory required to store keys is then $l \times (m + 1)$ where l is the key size.

B. Network Scalability

From construction, the total number of possible key rings when using the naive unital based scheme is $n = \frac{m^2 \times (m^3+1)}{(m+1)} = m^2 \times (m^2 - m + 1)$, this is then the maximum number of supported nodes.

C. Session Key Sharing Probability

When using the basic unital mapping, we know that each key is used in exactly m^2 key rings among the $m^2 \times (m^2 - m + 1)$ possible key rings. Let us consider two nodes u and v randomly selected. The node u is pre-loaded with a key ring KR_u of $m+1$ different keys. Each of them is contained in $m^2 - 1$ other key rings. Knowing that each two keys occur together in exactly one block, we find that the blocks containing two different keys of KR_u are completely disjoint. Hence, each node shares exactly one key with $(m + 1) \times (m^2 - 1)$ nodes among the $m^2(m^2 - m + 1) - 1$ other possible nodes, Then, the probability P_c of sharing a common key is :

$$\begin{aligned} P_c &= \frac{(m+1) \times (m^2-1)}{m^2 \times (m^2 - m + 1) - 1} \\ &= \frac{(m+1)^2 \times (m-1)}{(m-1) \times (m^3 + m + 1)} \\ &= \frac{(m+1)^2}{m^3 + m + 1} \end{aligned} \quad (1)$$

D. Summary and Discussion

The evaluation of this naive solution shows clearly that the basic mapping from unitals to key pre-distribution improves greatly the network scalability which reaches $O(k^4)$ compared to other schemes like SBIBD and trade ones having a scalability of $O(k^2)$ where k is the key ring size. Moreover, given a network size n this naive scheme allows to reduce the key ring size up to $\sqrt[4]{n}$. However, this naive solution degrades the key sharing probability which tends to $O(\frac{1}{k})$. In order to improve the key sharing probability of the naive unital based scheme while maintaining a good scalability improvement, we propose in the next section an enhanced construction for key management schemes based on unital design.

VI. A NEW SCALABLE UNITAL-BASED KEY PRE-DISTRIBUTION SCHEME FOR WSN

In this section, we present a new enhanced unital-based key pre-distribution scheme for WSN. In order to enhance the key sharing probability while maintaining high network scalability, we propose to build blocks using unital design and to pre-load each node with a number of blocks picked in a selective way.
Before the deployment step, we propose to generate blocks of a m order unital design, each block matches a key set. We propose then to pre-load each node with t $completely\ disjoint$ blocks, t is then a protocol parameter that we will discuss later. The aim of our construction is to enhance the key sharing probability between neighboring nodes and then decrease the average secure path length as we show later. We propose in algorithm 1 a random block distribution allowing to pre-load t disjoint blocks in each sensor node.

```
Generate B =< B_q >, key sets corresponding to
  blocks of a unital design of order m
foreach Node_i do
    KR_i = {}
    while (|KR_i| ≤ t(m+1)) do
        pick B_q from B
        if ((KR_i ∩ B_q) = ∅) then
            KR_i = KR_i ∪ B_q
            B = B − B_q
        end
    end
end
```

Algorithm 1: *A random approach of unital block pre-distribution in the enhanced unital-based scheme*

After the deployment step, each two neighbors exchange their key identifiers in order to determine common keys. Contrary to the basic approach, each two nodes may share more than one key when using the proposed construction. Indeed, each node is pre-loaded with t disjoint blocks which means that each two nodes share up to t^2 keys. If two nodes share one or more keys, we propose to compute the pairwise secret key as the hash of all their common keys concatenated to each other. The used hash function may be

SHA-1 [24] for instance. This approach allows to enhance the network resiliency since the attacker needs more overlap keys to break a secure link. Otherwise, when neighbors do not share any key, they should find a secure path composed of successive secure links.
The major advantage of this enhanced version is the improvement of the key sharing probability. As we will show later, this approach allows to achieve a high secure connectivity coverage since each node is pre-loaded with t disjoint blocks. Moreover, this approach increases resiliency through the composite pairwise secret keys witch reinforce secure links. In addition, we show that we maintain high network scalability compared to existing solutions although it remains lower than that of the naive version.

VII. Evaluation & comparison

In this section, we evaluate the performances of the enhanced unital-based scheme and compare it to the naive one and to the main existing solutions. We focus in comparisons on the design based schemes and mainly the SBIBD and the trade-based ones. We denote in what follows by NU-KP the naive unital-based key pre-distribution scheme proposed in section 5 and by t-UKP the enhanced unital-based scheme having t as parameter.

A. Storage Overhead of the t-UKP Scheme

When using a m order t-UKP scheme, each node is pre-loaded with $t(m+1)$ distinct keys. Indeed, from the construction, we notice that each t blocks of a key ring are completely disjoint and, then, do not intersect at any key. The memory required to store keys is then equal to $l \times t \times (m+1)$ where l is the key size.

B. Network scalability of the t-UKP scheme

Since each node is pre-loaded with t blocks from the $m^2 \times (m^2 - m + 1)$ possible blocks of a unital, it is obvious that the maximum number of key rings that we can reach is equal to $n = \frac{m^2}{t}(m^2 - m + 1)$. This is the ideal case when all unital blocks are used. When using the random pre-distribution presented in algorithm 1, we may generate a number of blocks slightly lower than this best value. We compute in what follows the minimum network size that can be supported using the random blocks distribution.

Lemma 1: Given $t \geq 2$, each set of $(t-1)(m+1)(m^2-1)+t$ blocks from a m order unital design contains at least one sub-set of t completely disjoint blocks.

Proof: As shown before, we know that each block of a unital design intersect with exactly $(m+1)(m^2-1)$ other blocks. We prove the proposition by induction: for $t = 2$, let T be a set of $(m+1)(m^2-1)+2$ blocks of a unital design of order m and let us assume that each two blocks of T intersect at one point. So, each block of T intersects with the $(m+1)(m^2-1)+1$ other blocks in T which contradicts the fact that each block intersects with exactly $(m+1)(m^2-1)$ blocks of the global unital. Hence the proposition is true for $t = 2$.
Let us now assume that the proposition is true at the order t and check whether it is for $t+1$. Let T be a set of $t(m+1)(m^2-1)+t+1$ blocks of a unital of order m. Since the proposition is true at order t, it exists at least one subset T_0 of t disjoint blocks in T. Each of these blocks intersects with exactly $(m+1)(m^2-1)$ other blocks. So the maximum possible number of blocks which intersect with T_0 is $t(m+1)(m^2-1)$. Hence, among the remaining $t(m+1)(m^2-1)+1$ blocks of $T - T_0$, there exists at least one block which does not intersect with any block of T_0. We deduce that T contains at least $t+1$ completely disjoint blocks.

■

Proposition 1: Using a unital design of order m, the algorithm 1 allows to generate at least $\frac{m^2(m^2-m+1)-((t-1)(m^2-1)(m+1)+t)}{t}$ key rings, , where t is the number of disjoint blocks in each key ring.

Proof: Using a unital design of order m, the number of the generated blocks is equal to $m^2(m^2-m+1)$. From the algorithm construction, we know that each key ring contains exactly t disjoint blocks. Following the lemma 1, we find that the algorithm generates at least $\frac{m^2(m^2-m+1)-((t-1)(m^2-1)(m+1)+t)}{t}$ key rings. ■

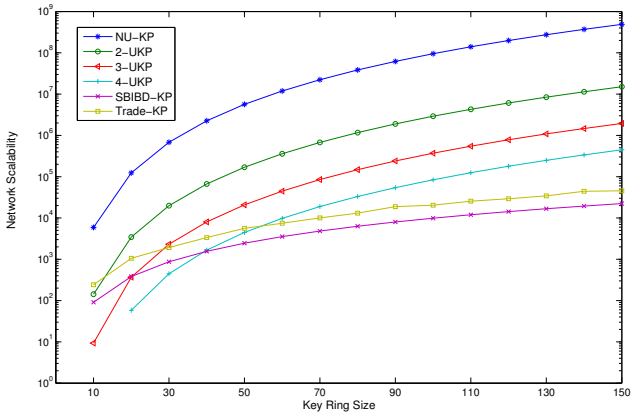

Fig. 2. Network scalability at equal key ring size

C. Comparison of network scalability at equal key ring size

We plot in figure 2 the scalability of the proposed unital based schemes and that of the SBIBD-KP and the Trade-KP ones in a logarithmic scale. The network scalability of the SBIBD scheme is computed as m^2+m+1 where m is the SBIBD design order and $m+1$ is the key ring size. We computed the salability of the Trade based scheme as $2q^2$ where q is the first prime power greater than the key ring size k. Indeed, the Ruj et al. trade construction generates $2q^2$ blocks of k keys each one such that $4 \leq k < q$ and q is a prime power. They propose to choose $q = O(k)$, this allows a achieve the best key sharing probability as we show later.

The figure shows that at equal key ring size, the naive unital key pre-distribution scheme allows to enhance greatly the scalability compared to the other schemes; for instance the increase factor reaches 10000 compared to the SBIBD-KP scheme when the key ring size exceeds 100. Moreover, the figure shows that the t-UKP schemes achieve a high network scalability. We notice that higher is t lower is the network scalability of t-UKP. Nevertheless, our solution gives better results than those of the SBIBD and the trade based solutions. For instance the 3-UKP scheme reaches a better network scalability than the SBIBD-KP and trade-KP ones when the key ring size is greater than 30. The increase factor reaches about 40 over the SBIBD-KP scheme and 20 over the trade-KP one when the key ring size exceeds 100.

Authors in [10], assess the network scalability of random schemes including the RKP and the Q-composite ones regarding to the desired key sharing probability p and to the network capacity to maintain secure links while some nodes are compromised. They defined for this second metric a threshold f_m called the *limited global payoff requirement*. Authors compute then the scalability depending on p and f_m, results for $p = 0.33$ and $f_m = 0.1$ show that the network scalability with a key ring size of 100 is about 300 for RKP scheme and between 600 and 700 when using Q-composite schemes. The scalability of the same schemes with a key ring size of 150 is respectively of about 800 and between 1000 and 1200. We can see clearly that our solutions allow to reach much better network scalability than the random schemes under the suggested parameters.

D. Key ring size at equal network size

We showed above that our approaches allows to address large networks compared to existing schemes at equal key ring size. In this subsection, we evaluate and compare the required key ring size when using the unital-based, the SBIBD-KP and the trade-KP schemes at equal network size. We compute for each network size the design order allowing to achieve the desired scalability and we deduce then the key ring size, the obtained results are reported in figure 3. The figure shows that at equal network scalability, the naive unital based scheme allows to reduce extremely the key ring size and then the storage overhead. Indeed the enhancement factor over the SBIBD reaches 20. When using the t-UKP schemes, the figure shows that higher is t, higher is the required key ring size. However, this value remains significantly lower than the required key ring size of the SBIBD-KP and Trade-KP schemes.

E. Session Key Sharing Probability of the t-UKP Scheme

We compute in this subsection the key sharing probability of the enhanced unital-based scheme and we show the improvement over the naive version.

Proposition 2: Given $t \geq 2$, using a m order t-UKP scheme, the key sharing probability between any two nodes P_c is given by:

$$P_c = 1 - (1 - \frac{(m+1)^2}{m^3+m+1})^{t^2} \quad (2)$$

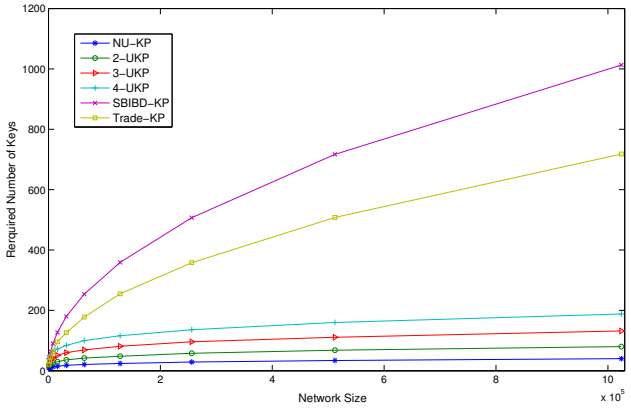

Fig. 3. Required key ring size at equal network size

Proof: Let us consider two randomly selected nodes u and v. Each node is pre-loaded with a key ring containing t disjoint unital blocks:
$KR_u = \{B_{u,1} \cup B_{u,2} \cup ... \cup B_{u,t}\}$ and
$KR_v = \{B_{v,1} \cup B_{v,2} \cup ... \cup B_{v,t}\}$
Following equation (1), we find that the probability that two blocks $B_{u,p}$ and $B_{v,q}$ share one key is $\frac{(m+1)^2}{m^3+m+1}$. The probability that they don't share any key is then $(1 - \frac{(m+1)^2}{m^3+m+1})$.

Since all blocks of node u as well as those of node v are completely disjoint thanks to the proposed construction, the probability that the two nodes u and v don't share any key is then given by:

$$\begin{aligned} P(KR_u \cap KR_v = \emptyset) &= \prod_{p=1}^{t} \left(\prod_{q=1}^{t} P(B_{u,p} \cap B_{v,q} = \emptyset) \right) \\ &= \prod_{p=1}^{t} \left(\prod_{q=1}^{t} (1 - \frac{(m+1)^2}{m^3+m+1}) \right) \\ &= (1 - \frac{(m+1)^2}{m^3+m+1})^{t^2} \end{aligned}$$

The probability that two nodes share at least one key is then:

$$P_c = 1 - (1 - \frac{(m+1)^2}{m^3+m+1})^{t^2}$$ ■

F. Comparison of Session Key Sharing Probability

We compare in this subsection, the session key sharing probability of the unital-based schemes to those of the SBIBD-KP and the Trade-KP ones. We prove in appendix A that the key sharing probability of the Ruj et al. Trade-KP scheme [20] is equal to $\frac{k(k-1)}{4q^2}$ where $4 \leq k < q$ and q is a prime power. k is the key ring size and we chose q to be the first prime power greater than k which ensures the best key sharing probability.

We plot in figure 4 the obtained results. The figure shows that the naive unital-based key pre-distribution provides a bad secure connectivity which decreases significantly when the key ring size increases. Otherwise, the obtained results show that the key sharing probability is significantly enhanced when t is high. The figure shows that higher is t, better is the key sharing probability. Indeed, loading nodes

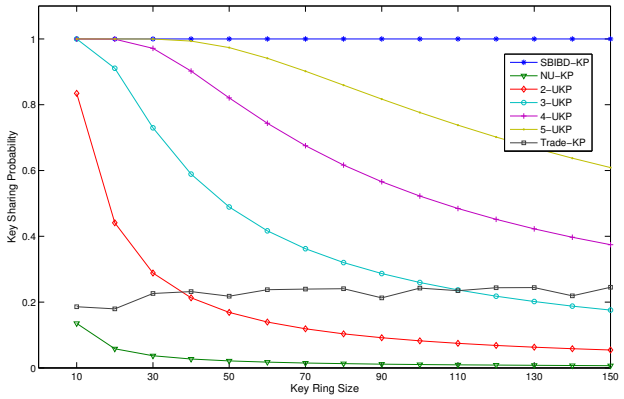

Fig. 4. Session key sharing probability at equal key ring size

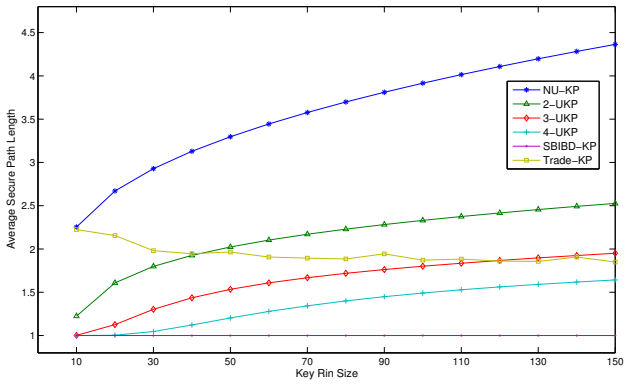

Fig. 5. Average secure path length at equal key ring size

with many blocks from unital design allows to increase significantly the key sharing probability. For instance 5-UKP allows to achieve key sharing probability of 0.6 when the key ring size is 150, the same key sharing probability is reached by the enhanced scheme with $t = 4$ when the key ring size is equal to 80.

Although the enhanced unital-based scheme increases significantly the network scalability as shown before, the resulting good key sharing probability remains lower compared to SBIBD scheme which guarantee a total connectivity. However, our scheme allows to attend a total secure connectivity thanks to the secure path establishment. We show that the average secure path length remains good when using the enhanced version.

G. Average secure path length

As explained before, when two neighboring nodes have no common keys, we propose to establish a secure path composed of successive secure links. In this subsection, we study the average secure path length when using different key pre-distribution schemes including the proposed unital-based solutions. For this purpose, we refer to the results given in [25] in order to construct the deployment model. Authors give the necessary and sufficient condition to ensure a connected and covered network when deploying sensor nodes. Following this study, we conducted our simulations by deploying n nodes in a unit square region while $r = \sqrt{\frac{log(n)}{n}}$ where r is the communication range and n is network size.

The figure 5 shows that the naive unital based scheme gives a relatively high average secure path length which is between 2 and 4 when the key ring size is between 10 and 120 and which is greater than 4 when the key ring size exceeds 120. However, the results show that the enhanced version with $t = 2$, 3 and 4 gives better results than the basic one, For instance the 3-UKP scheme allows to reach a low secure path length lower than two. The figure shows that higher is t lower is the average secure path length. Moreover, we notice that when the key ring size is lower than 150, the enhanced version gives better results than the trade-KP ones when t is greater than or equal to 3.

H. Choice of the t value

As shown before, the pre-distribution of t unital blocks in each node instead of one allows at the same time enhancing the key sharing probability and computing composite pair-wise secret keys witch reinforce secure links. On the other hand, the use of the enhanced version multiplies the storage overhead and decreases the network scalability over the naive version. However, we notice that these performance metrics remains good compared to existing solutions as we confirm in comparison section. Indeed the storage overhead is about $O(t \times m) = O(m)$ while the network scalability tends to $O(\frac{m^4}{t}) = O(m^4)$.

The choice of the t value depends then on the application requirement in order to obtain the best tradeoff. Indeed, when we do not need to establish a secure link between each pair of nodes or when the length of secure paths is not a major concern, low values of t can be chosen. For instance, in many-to one WSN where the key sharing requirement is reduced to the child-parent relationship, the proposed scheme with low values of t can be efficiently used to construct a secure tree rooted at the sink. In these situations we can reach an extremely high scalable deployment thanks to the low value of t. On the other hand, when the key sharing probability and the length of secure paths are major concerns in the application, t should be given a high value which allows to ensure a good key sharing probability. The t value should be then computed depending on the desired key sharing probability while taking into account the memory size and the required network size.

VIII. Conclusion & Future Work

We proposed, in this paper, a new highly scalable key pre-distribution scheme for WSN. We make use, for the first time, of the unital design theory. We showed that a basic mapping from unitals to key pre-distribution allows to achieve an extremely high network scalability while degrading the key sharing probability. We proposed then an enhanced unital-based construction which gives birth to

a new key management scheme providing high network scalability and good key sharing probability. We conducted analytic calculation and intensive simulations to compare our solutions to existing ones which showed that our approach enhances significantly the network scalability when providing good overall performances. As future work, we plan to deepen the analysis of our parameter choice in order to suggest values given the best tradeoff. In addition, we attend to analyze more network performances of our solution like the network resilience against node capture attacks.

IX. Acknowledgement

This work is made as part of the Picardie regional project under reference I159C. The authors wish to thank the Picardie regional council in France and the European Regional Development Fund (ERDF) for funding and supporting this project.

Appendix A
Key sharing probability of the Trade-KP scheme

We compute in what follows the key sharing probability of the trade-based scheme proposed in [20]. We recall that the Ruj et al. trade construction presented before allows to generate two sets T_1 and T_2 of q^2 key rings each one. Each key ring contains k keys.

Following the 2-steiner trade property we know that each pair of elements (keys) occurs at most once in T_1 and at most once in T_2. Authors prove that their construction satisfy this property. This means that each two key rings from T_1 have no pair of shared keys and each two key rings from T_2 have no pair of shared keys also. Indeed, two keys rings can share a pair of keys if and only if they belongs to distinct sets T_i.

Let us now take a key ring from T_1, we know that it contains exactly k keys and so $\binom{k}{2}$ possible pair of keys. Following the construction, we find that it shares each couple of keys with exactly one other key ring from T_2. So each two key rings selected from T_1 and T_2 respectively share a couple of keys with a probability $\frac{\binom{k}{2}}{q^2}$

In the general case, if we consider two key rings KR_i and KR_j randomly selected from $T_1 \cup T_2$, the probability that they share a pair of keys which is the probability that the corresponding nodes can establish a secure link is given by:

$$\begin{aligned} Pc &= P((KR_i, KR_j) \in (T_1 \times T_1 \cup T_2 \times T_2)) \times 0 \\ &+ P((KR_i, KR_j) \in (T_1 \times T_2 \cup T_2 \times T_1)) \times \frac{\binom{k}{2}}{q^2} \\ &= \frac{1}{2} \times 0 + \frac{1}{2} \times \frac{\binom{k}{2}}{q^2} \\ &= \frac{k(k-1)}{4q^2} \end{aligned}$$

References

[1] Yun Zhou, Yuguang Fang, and Yanchao Zhang. Securing wireless sensor networks: A survey. *IEEE Communications Surveys and Tutorials*, 10(1-4):6–28, 2008.

[2] An Liu and Peng Ning. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks. In *Proceedings of the 7th international conference on Information processing in sensor networks*, IPSN '08, pages 245–256, Washington, DC, USA, 2008. IEEE Computer Society.

[3] R. Watro, D. Kong, S. Cuti, C. Gardiner, C. Lynn, and P. Kruus. Tinypk: securing sensor networks with public key technology. In *In SASN 04: Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, pages 59–64. ACM Press, 2004.

[4] Nils Gura, Arun Patel, Arvinderpal Wander, Hans Eberle, and Sheueling Chang Shantz. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *CHES*, pages 119–132, 2004.

[5] J. Zhang and V. Varadharajan. Wireless sensor network key management survey and taxonomy. *Journal of Network and Computer Applications*, 33(2):63 – 75, 2010.

[6] K. Kifayat, M. Merabti, Q. Shi, and D. Llewellyn-Jones. Security in wireless sensor networks. In *Handbook of Information and Communication Security*, pages 513–552. 2010.

[7] D. Snchez and H. Baldus. Key management for mobile sensor networks. In *Secure Mobile Ad-hoc Networks and Sensors*, volume 4074 of *Lecture Notes in Computer Science*, pages 14–26. Springer Berlin / Heidelberg, 2006.

[8] A. Ouadjaout, M. Bagaa, A. Bachir, Y. Challal, N. Lasla, and L. Khelladi. *Encyclopedia on Ad Hoc and Ubiquitous Computing*, chapter Information Security in Wireless Sensor Networks, pages 427–472. World Scientific, 2009.

[9] L. Eschenauer and V.D. Gligor. A key-management scheme for distributed sensor networks. In *ACM CCS '02*, pages 41–47, 2002.

[10] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IEEE SP '03*, 2003.

[11] W. Du, J. Deng, Y. Han, S. Chen, and P. Varshney. A key management scheme for wireless sensor networks using deployment knowledge. In *IEEE INFOCOM'04*, pages 586–597, 2004.

[12] D. Liu and P. Ning. Establishing pairwise keys in distributed sensor networks. In *ACM CCS '03*, pages 52–61, 2003.

[13] R. Blom. An optimal class of symmetric key generation systems. In *Proceedings of the Eurocrypt 84 Workshop on Advances in Cryptology: Theory and Application of Cryptographic Techniques.*, pages 335–338. Springer-Verlag, 1985.

[14] Z. Yu and Y. Guan. A robust group-based key management scheme for wireless sensor networks. In *Wireless Communications and Networking Conference*, pages 1915–1920. IEEE, 2005.

[15] Fang Liu, Xiuzhen Cheng, Liran Ma, and Kai Xing. Sbk: A self-configuring framework for bootstrapping keys in sensor networks. *IEEE Transactions on Mobile Computing*, 7(7):858–868, 2008.

[16] Taehwan Choi, Hrishikesh B. Acharya, and Mohamed G. Gouda. The best keying protocol for sensor networks. In *WOWMOM*, pages 1–6, 2011.

[17] S. Zhu, S. Setia, and S. Jajodia. Leap: efficient security mechanisms for large-scale distributed sensor networks. In *ACM CCS '03*, pages 62–72, 2003.

[18] S. A. Çamtepe and B. Yener. Combinatorial design of key distribution mechanisms for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 15:346–358, April 2007.

[19] Sushmita Ruj and Bimal Roy. Key predistribution using combinatorial designs for grid-group deployment scheme in wireless sensor networks. *ACM Trans. Sen. Netw.*, 6:4:1–4:28, January 2010.

[20] Sushmita Ruj, Amiya Nayak, and Ivan Stojmenovic. Fully secure pairwise and triple key distribution in wireless sensor networks using combinatorial designs. In *INFOCOM*, pages 326–330, 2011.

[21] E.F. Assmus and J.D. Key. *Designs and their codes*. Cambridge tracts in mathematics. Cambridge University Press, 1992.

[22] Anton Betten, Dieter Betten, and Vladimir D. Tonchev. Unitals and codes. *Discrete Mathematics*, 267(1-3):23–33, 2003.

[23] Jennifer D. Key. Some applications of magma in designs and codes: Oval designs, hermitian unitals and generalized reed-muller codes. *J. Symb. Comput.*, 31(1/2):37–53, 2001.

[24] National Institute of Standards and Technology. Secure hash standard. *Federal Information Processing Standards Publication*, 1995.

[25] Sanjay Shakkottai, Rayadurgam Srikant, and Ness B. Shroff. Unreliable sensor grids: coverage, connectivity and diameter. *Ad Hoc Networks*, 3(6):702–716, 2005.