Using MDS Codes for the Key Establishment of Wireless Sensor Networks*

Jing Deng[1] and Yunghsiang S. Han[2]

[1] Department of Computer Science,
University of New Orleans,
New Orleans, LA 70148, USA
jing@cs.uno.edu
[2] Graduate Institute of Communication Engineering,
National Taipei University, Sanhsia, Taipei, 237 Taiwan , R.O.C
yshan@mail.ntpu.edu.tw

Abstract. Key pre-distribution techniques for security provision of Wireless Sensor Networks (WSNs) have attracted significant interests recently. In these schemes, a relatively small number of keys are randomly chosen from a large key pool and loaded on the sensors *prior to* deployment. After being deployed, each sensor tries to find a common key shared by itself and each of its neighbors to establish a link key to protect the wireless communication between themselves. One intrinsic disadvantage of such techniques is that some neighboring sensors do not share any common key. In order to establish a link key among such neighbors, a multi-hop secure path may be used to deliver the secret. Unfortunately, the possibility of sensors being compromised on the path may render such establishment process insecure.

In this work, we propose and analyze an Incremental Redundancy Transmission (IRT) scheme that uses the powerful Maximum Distance Separable (MDS) codes to address the problem. In the IRT scheme, the encoded secret link key is transmitted through multiple multi-hop paths. To reduce the total information that needs to be transmitted, the redundant symbols of the MDS codes are transmitted only if the destination fails to decode the secret. One salient feature of the IRT scheme is the flexibility of trading transmission for lower information disclosure. Theoretical and simulation results are presented to support our claim.

1 Introduction

Wireless Sensor Networks (WSNs) have attracted significant interests from the research community due to their potentials in a wide range of applications such as environmental sensing, battlefield sensing, and hazard leak detection. The security problem of these WSNs are important as the sensors might be deployed

---

to unfriendly areas. When any of the sensors is compromised or captured, the information on the sensor is disclosed to the adversary and its operation may become under the control of the adversary.

In order to secure the communication between a pair of sensors, a unique key is needed. Since public/private (asymmetric) keys consume large amount of system resource, the secret (symmetric) key technique is preferred in WSNs. However, distribution of a secret key for every possible communication link is non-trivial due to the large number of sensors and the limited on-board memory size. To this trend, key pre-distribution techniques have been proposed and studied [1, 2, 3, 4]. These techniques allow the sensors to randomly pick a relatively small number of keys from a large key pool and two neighboring nodes[1] then try to find a common key that is shared by themselves.

Due to the randomness of the key selection process in key pre-distribution, some communication links do not have any common key shared by the two neighboring nodes. In [2], a secret link key delivery technique using a multi-hop secure path was proposed: one of the two neighboring nodes finds a multi-hop secure path toward the other node.[2] Each pair of neighboring nodes on the secure path share at least a common key, which could be different throughout the path. Then a secret link key is generated from the source node and sent toward the destination through the secure path.

Such a multi-hop secure path scheme works quite well when all sensor nodes forward the secret key honestly and none of the nodes on the path is compromised. However, the scheme has security problems if any of the nodes is compromised or captured by the adversary. Such a compromise affects the multi-hop secure path scheme in the following way: 1) since the secret link key is decrypted and re-encrypted by each sensor on the path, it may be disclosed to the adversary; and 2) the adversary can modify or drop the information passing through.

In this work, we address the problem of compromised sensors modifying and eavesdropping the information passing through such multi-hop paths. We use the powerful Maximum Distance Separable (MDS) codes to develop the Incremental Redundancy Transmission (IRT) scheme to provide protection for information delivery. Our analysis and simulation results show that the proposed technique is highly efficient. Note that, in [2], Chan, Perrig, and Song proposed a multi-path reinforcement scheme that is similar to our scheme. However, the multi-path reinforcement scheme only concerns the information disclosure to the adversary but not information modification by the adversary.

The paper is organized as follows: in Section 2, we overview related work. The secret link key delivery problem is formulated in Section 3. In Sections 4 and 5, we present the IRT scheme and our analysis. The performance evaluation results are provided in Section 6. We summarize and conclude this work in Section 7, stating some possible future directions.

---

[1] In this work, we use "sensors", "nodes", and "sensor nodes" interchangeably.

[2] Note that the two communicating nodes are physical neighbors. Such a small geographical separation between the source and the destination enables prompt and efficient secret verifications.

2   Related Work

Reference [1] proposed a random key establishment technique for WSNs. In this technique, each sensor is pre-loaded a number of keys that are randomly selected from a large key pool. After deployment, two neighbors can establish a secure communication if they share a common key. Otherwise, they need to exchange a secret key via a multi-hop secure path. Reference [2] extended the technique into q-composite random key establishment technique which forces two neighbors to establish a secure communication only when they share q common keys, where $q \geq 2$. Based on [1], two similar random key pre-distribution techniques that used multi-space key pool to drastically improve network resilience and memory usage efficiency were developed independently [3, 4].

A multi-path key reinforcement technique was proposed in [2] to enable two nodes to establish secure communication even if they do not share enough common keys (with the use of the q-composite technique). These two neighbors first identify all secure paths between themselves. Then one node generates a set of random numbers (of the same size) for all the paths and send each number to the other node through each of the paths. After the destination receives all the numbers, it exclusive-ORs all of them to obtain the secret link key. The multi-path key reinforcement scheme significantly improves the protection of the secret link key from being disclosed to the adversary. However, the scheme fails if any of the paths is compromised by the adversary and the number is modified or dropped.

In [5, 6], combinatorial set was used to distribute keys to sensors *prior to* deployment. Such a deterministic combinatorial set technique allows each key in the key pool to be assigned to a constant number of sensor nodes. Therefore, the number of nodes each sensor shares a common key is fixed.

A Secure Routing Protocol (SRP) was proposed in [7] to send additional information to protect routing information being dropped. To combat the topology instability problem in wireless networks, a multi-path routing scheme was proposed and investigated in [8]. The scheme allows the sender to add extra overhead to each packet that is to be transmitted over multiple paths. The goal is to find the optimal way to fragment the packet into smaller blocks and deliver them over multiple paths. The focus of [7, 8] is on the problem of missing some of the messages but not modification of them.

In [9], an efficient information dispersal mechanism was developed to provide security, load balance, and fault tolerance for communication networks. Reed-Solomon codes were used to recover link faults and to provide security. However, all redundancy were sent along with the information symbols, increasing transmission overhead significantly.

MDS codes have been used in the Automatic-Repeat-reQuest (ARQ) protocols to reduce the transmission overhead on communication systems [10, 11]. In [10], a family of MDS codes, Reed-Solomon codes, was used in a type-II hybrid ARQ protocol. In the first transmission, a relatively high rate Reed-Solomon code with fewer redundancy is used. When an additional transmission is needed, only the redundant symbols are sent. With such a technique, the overall code

rate is reduced. This scheme increases the system throughput by reducing the transmission overhead. In [11], punctured MDS codes were used for the type-II hybrid ARQ protocol and a modified version (with fewer decoding operations) of the scheme proposed in [10] was presented.

3   Problem Formulation

We explain the link key establishment problem in WSNs in more details in this section. The key pre-distribution schemes such as [1, 3, 4, 12] provide memory-efficient and resilient ways to establish secret link keys for a portion of potential communication links. The rest of the communication links need to establish their secret keys by other means such as multi-hop delivery.[3]
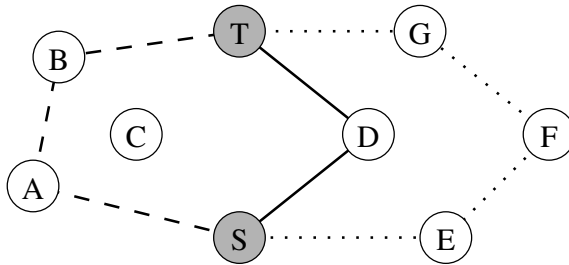

Fig. 1. Illustrations of multi-hop key establishment

A few sensor nodes are shown in Fig. 1. Each line segment connecting two nodes represents that these two nodes share at least a common key, e.g., nodes E and F share at least a key. Note that nodes S and T do not share any common key. Now assume that nodes S and T, which are physical neighbors, need to establish a secure communication that requires a secret link key. As suggested in [1, 2, 3], in order to establish a secret link key between nodes S and T, a multi-hop secure path may be used to deliver the secret key. For example, node D may be used to relay the secret key between nodes S and T. Since only one multi-hop path is used in the key delivery, we term it the Single Path (SP) scheme.

The SP scheme may be summarized as follows: when node S needs to establish a secret link key, K_{link}, with node T, node S finds a path $S - N_1 - N_2 - \cdots - N_h - T$, where S shares a key with N_1, N_1 shares a key (could be different) with N_2, etc.. Then node S encrypts K_{link} with the secret key shared by itself and node N_1 and sends the encrypted message to node N_1. Node N_1 decrypts K_{link} using the common secret key shared with node S. Then node N_1 sends K_{link} to

---

[3] We assume that some sensor nodes may be compromised even during the WSN initialization process and the adversary is able to decrypt the recorded information after it compromises some sensor nodes later on. An assumption of secure initialization process will make the key establishment issue trivial.

node N_2 using the same technique. This process continues until K_{link} reaches the destination, node T. Some examples of such multi-hop paths in Fig. 1 are $S - D - T$, $S - A - B - T$, and $S - E - F - G - T$.

Note that the secret link key K_{link} needs to be decrypted and then re-encrypted by each of the sensor nodes on the multi-hop path. This leads to the following problem:

Problem Statement: *In key pre-distribution schemes for WSNs, some neighboring sensors do not share any common key. Their secret link key needs to be established through multi-hop secure path. However, when any of the sensors on the multi-hop secure path is compromised or captured by the adversary, the secret link key is disclosed. A compromised sensor may also modify or drop the key information passing through itself. How do we provide a fault tolerant mechanism to send the secret link key between two physical neighbors efficiently and securely?*

In Section 4, we introduce the powerful Maximum Distance Separation (MDS) codes and use it in our Incremental Redundancy Transmission (IRT) scheme to solve this problem.

4   The Incremental Redundancy Transmission (IRT) Scheme

4.1   MDS Codes

We first review the MDS codes [10, 11] that will be used in the IRT scheme. Let Hamming distance between two vectors (codewords) be the number of distinct positions between two vectors. An (n, k, d_{min}) MDS code is a linear block code whose minimum Hamming distance d_{min} between any pair of distinct codewords must satisfy $d_{min} = n - k + 1$, where n is the code length and k is the dimension of the code. Therefore, each codeword in the $(n, k, n - k + 1)$ MDS code has exactly n symbols among which there are k information symbols. Usually, the extra $n - k$ symbols are called parity checks or redundancy of the code. Furthermore, an $(n, k, n - k + 1)$ MDS code will be able to recover any t errors if

$$t \leq \left\lfloor \frac{n - k}{2} \right\rfloor.$$

MDS codes are optimal in the sense that they provide the largest possible Hamming distance between codewords and hence can correct most number of errors. The most famous family of MDS codes are Reed-Solomon codes. Efficient decoding algorithms for MDS codes have been studied extensively in [13]. The MDS codes have several nice properties that make them very useful. Two of such properties are given as follows without proof.

Property 1. Punctured (shortened) MDS codes are MDS.

A code is punctured by deleting parity symbols from each codeword in the code and a code is shortened by deleting information symbols from each codeword in

the code. An $(n, k, n - k + 1)$ MDS code can be punctured (shortened) to an $(n - j, k, n - j - k + 1)$ $((n - j, k - j, n - k + 1))$ MDS code by deleting j parity (information) symbols from each codeword.

Property 2. Any k coordinates of an MDS code can be used as information symbols.

According to this property, by knowing any k symbols of a codeword of an MDS code, one can recover other $n - k$ symbols for this codeword.

4.2   Multi-hop Paths

Before we present the IRT scheme, we discuss the path selection process and its effect on IRT. In this work, we assume that a source node identifies m multi-hop paths between itself and the destination. Such m paths could be chosen from the node-disjoint paths, where none of the paths has a common node besides the source and the destination [14]. Another option is to allow the source node to select randomly among the available paths. The result is that some paths may have common nodes and thus the security performance worsens. The benefit of such a selection technique is that it does not rely on the availability of node-disjoint multi-hop paths and eliminates the cost of identifying such paths.

As suggested in [2], it is always beneficial to choose short multi-hop paths instead of long multi-hop paths. As the length of a multi-hop path increases, the possibility of path compromise is higher. On the other hand, the number of short multi-hop paths is usually smaller than that of long multi-hop paths. We evaluate the values of m under various network conditions and the effect of such m paths on the security performance of our scheme in Section 6.

4.3   The Incremental Redundancy Transmission (IRT) Scheme

Let $\boldsymbol{c} = (c_0, c_1, \ldots, c_{k-1}, c_k, \ldots, c_{n-1})$ be a codeword of an (n, k) MDS code over $GF(q)$, where $c_i \in GF(q)$, $0 \le i \le n - 1$, is a symbol. Here $c_0, \ldots, c_{k-1}$ are the information that the source needs to send to the destination. Assume that the secret link key between them is of length $\gamma < k$ and can be generated by a function f of these information if the destination decodes this codeword correctly. The design goal of our scheme is to tolerate up to e compromised paths. Since an (n, k) MDS code can recover up to $(n - k)/2$ errors, n should satisfy

$$n \ge k + 2e , \tag{1}$$

where we have implicitly assumed that $n - k$ is even. Since the identities of the compromised paths are unknown to the source node, the best strategy for the source is to split data evenly and send to all available paths.

A brief outline of the IRT scheme is given as follows: let $r_0 = k$ be the number of symbols sent by the source in the original transmission. If the destination receives all information correctly and re-generates the secret key sent by the

source, our scheme terminates. Otherwise, the source sends r_1 symbols in its first additional transmission. If the destination succeeds in secret key re-generation, our scheme stops. Otherwise, the same process continues until the n symbols are exhausted.

Let $r_1, r_2, \ldots, r_e$ be the numbers of extra symbols sent out by the source in each additional transmission (the values of r_i, $1 \leq i \leq e$, will be determined in Section 5). Note that, in our scheme, the source only needs to send out up to $\sum_{i=0}^{t} r_i$ symbols when t paths are compromised. Let $\boldsymbol{y} = (y_0, y_1, \ldots, y_{\ell-1})$ be the corresponding received vector at the destination when the source has sent out ℓ symbols up to now.

The IRT scheme is given as follows:

1. The source first encodes k symbols, $\boldsymbol{c}_0 = (c_0, c_1, \ldots, c_{k-1})$, into a codeword with n symbols, $\boldsymbol{c} = (c_0, c_1, \ldots, c_{k-1}, c_k, \ldots, c_{n-1})$, where $n = \sum_{i=0}^{e} r_i = k + \sum_{i=1}^{e} r_i$ and $r_0 = k$. Initialize $i = 0$, $b = 0$, and $s = r_i - 1$.
2. The source transmits r_i symbols specified by b and s ($\boldsymbol{c}_i = (c_b, c_{b+1}, \ldots, c_s)$) evenly along the m paths. If r_i/m is not an integer, the last path will transmit less symbols.
3. Assume that the destination receives all symbols from the m paths as $\boldsymbol{y}_i = (y_b, y_{b+1}, \ldots, y_s)$.[4] The destination appends $\boldsymbol{y}_i$ to all the previously received symbols to form a longer codeword. Then it tries to decode this codeword in order to obtain the k symbols. If the decode process fails due to more than i errors occur, then go to Step 4 directly; otherwise, it verifies this result with the source through the challenge-response technique (recall that the source and the destination are physical neighbors). If the re-generated secret link key is verified, the transmission of the secret link key has succeeded; Otherwise, goes to Step 4.
4. If $i = e$, then the key establishment fails due to too many compromised paths. Otherwise, the destination asks for another round of additional transmission.
5. The source sets $i = i + 1$, $b = s + 1$, $s = s + r_i$, and repeats Step 2.

Therefore, compared with [10, 11], which have only one additional transmission, the IRT scheme sends multiple retransmissions when necessary.

5   Analysis

In this section, we derive the values of r_i, $0 \leq i \leq e$, for the IRT scheme. We will also discuss the security performance of the IRT scheme and the SP scheme, which sends the secret link key through a single multi-hop path.

5.1   Selections of $r_1, r_2, \ldots, r_e$

The values of $r_1, r_2, \ldots, r_e$ can be determined as follows. In order to reduce the total number of symbols to be transmitted, in each transmission we should

---

[4] The source can notify the destination the number of transmitted symbols over plain text. Therefore, the event of symbols being dropped is similar to that of symbols being modified along the multi-hop paths.

add as few as possible redundancy that can correct one more error.[5] Therefore, noticing that r_1 symbols are added in order to correct the errors caused by one compromised path, we can determine r_1 as

$$\left(\frac{r_1}{m} + \frac{k}{m}\right) \leq \frac{r_1}{2} . \tag{2}$$

The left side of (2) is the total number of errors introduced by the compromised path. The right side of (2) is the error correction capability due to the transmission of the additional r_1 symbols.

Taking the smallest integer that satisfies the above inequality, we have

$$r_1 = \left\lceil \frac{2k}{m-2} \right\rceil . \tag{3}$$

In general, the value of r_ℓ, where $1 \leq \ell \leq e$, must satisfy the following inequality

$$\frac{\ell}{m}\left(k + \sum_{i=1}^{\ell} r_i\right) \leq \frac{1}{2}\sum_{i=1}^{\ell} r_i . \tag{4}$$

In order to reduce the total number of message transmissions, we choose the smallest r_ℓ that satisfies the above inequality. Therefore,

$$\begin{aligned} r_\ell &= \left\lceil \frac{2\ell k}{m-2\ell} - \sum_{i=1}^{\ell-1} r_i \right\rceil \\ &= \left\lceil \frac{2\ell k}{m-2\ell} \right\rceil - \sum_{i=1}^{\ell-1} r_i . \end{aligned} \tag{5}$$

(5) can be further rearranged to

$$r_\ell = \left\lceil \frac{2\ell k}{m-2\ell} \right\rceil - \left\lceil \frac{2(\ell-1)k}{m-2(\ell-1)} \right\rceil , \tag{6}$$

when $1 \leq \ell \leq e$.

Based on (5), when there are ℓ compromised paths between the source and the destination, the total additional symbols that should be transmitted is

$$\sum_{i=1}^{\ell} r_i = \left\lceil \frac{2\ell k}{m-2\ell} \right\rceil. \tag{7}$$

Since e is the maximum number of errors that can be corrected by the IRT scheme, the set of $r_1, r_2, \ldots, r_e$ should satisfy:

$$\sum_{i=1}^{e} r_i \leq n - k ,$$

---

[5] We neglect the overhead of sending such symbols in our analysis. Inclusion of overhead such as MAC layer headers and physical layer headers may affect the performance of our scheme.

which leads to (based on (7))

$$\left\lceil \frac{2ek}{m - 2e} \right\rceil \leq n - k .$$

Therefore, e should satisfy

$$e < \frac{n - k}{n} \cdot \frac{m}{2} . \tag{8}$$

As a point of reference, when $n = 1024$, $k = 256$, $m = 15$, the maximum value of e is 5 due to (8). The array r_i, $0 \leq i \leq e$, is $\{256\ 39\ 54\ 77\ 122\ 220\}$.

5.2   Information Disclosure

We start our discussions on the security performance of information disclosure by presenting the attack model in the following: the adversary takes control of some compromised sensors and collects all information passing through them. An intelligent adversary may decide not to modify the information that is passing through in order to maximize the benefit of its eavesdropping effort.

We further assume that there are m_x paths that are compromised by the adversary among the m available paths.[6] We evaluate the *secret disclosure probability*, p_x, which is defined as the probability of disclosing enough symbols to the adversary so that it can obtain the key with relative ease.

In the SP scheme, the γ symbols are transmitted through one randomly chosen path among the m available paths. The secret disclosure probability can be calculated as

$$p_x^{(SP)} = \frac{m_x}{m} . \tag{9}$$

When the IRT scheme is used, the source transmits only $r(0) = k$ symbols and the destination gets all of these symbols successfully because no information is modified. Since such k symbols are transmitted through the m paths evenly, each path sends $\frac{k}{m}$ symbols. For a fair comparison between the IRT scheme and the SP scheme, we define the secret disclosure probability as the probability of at least αk symbols being disclosed to the adversary, where $0 < \alpha \leq 1$. Therefore, the secret disclosure probability can be calculated as

$$p_x^{(IRT)} = \frac{[\text{Number of cases where at least } \alpha k \text{ symbols are disclosed}]}{[\text{Total number of cases}]} . \tag{10}$$

The value of α depends largely on how the γ symbols of the secret link key information are encoded into the k symbols, i.e., it depends on the selection of function f in the scheme.

---

[6] We use a different variable than e in order to distinguish the two different kinds of compromises: information modification and information disclosure.

6   Performance Evaluation

Simulations were performed in Matlab to evaluate the efficiency of the proposed scheme. Unless specified otherwise, our simulations were set up with the following parameters: we randomly place $N = 400$ nodes on a square area of 1000 m by 1000 m. The radio transceiver range is 100 m. The MDS code is assumed to be $(n, k) = (1024, 256)$. We investigate the performance of the IRT scheme and the SP scheme for different γ (instead of varying (n, k)).

In Fig. 2, we show the number of paths with secure connections that are exactly h-hops from a source to a destination (assuming that they do not share a common key). The average number of paths is presented corresponding to various probabilities of any two neighboring nodes sharing a common key, p_{local}. We also present the number of paths for a similar network with half of the nodes, for comparison purposes. As can be observed from Fig. 2, the number of available paths increases with local connectivity, p_{local}. When nodal density increases, there are more paths as well. The number of h-hop paths also increases with h. Note that these paths may have common nodes besides the source and the destination.

When there are compromised nodes on the paths that are used to deliver the secret link key information and these compromised nodes modify the passing information, extra symbols need to be transmitted. Figure 3 shows the average number of symbols that need to be transmitted in order to allow the destination to re-generate the secret key. We use all of the available 2- and 3-hop paths in the IRT scheme. The SP scheme randomly chooses one out of these paths to send the secret link key. In Fig. 3, we fix (n, k) for the IRT scheme but vary γ. Therefore, the number of transmitted symbols in the SP scheme lowers as γ decreases. The relative transmission cost of the IRT scheme increases as γ
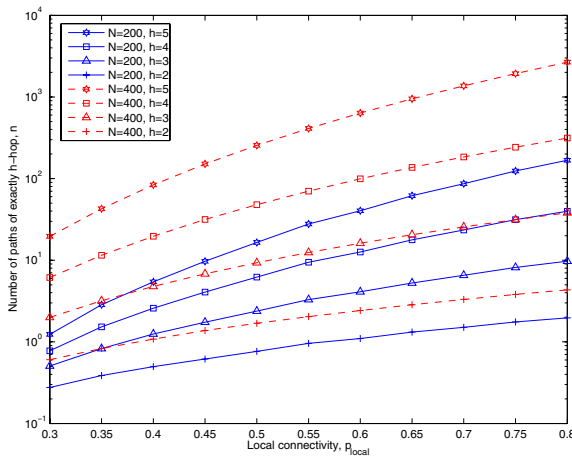

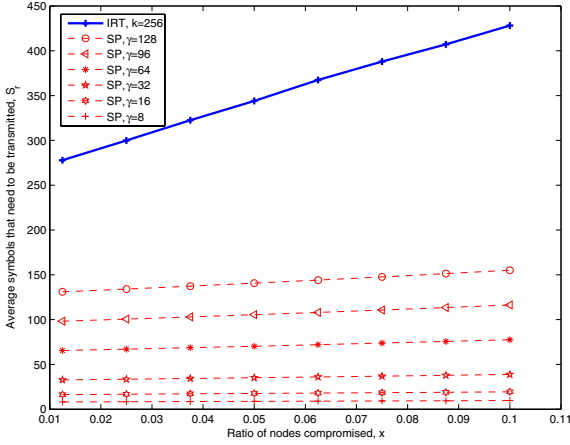**Fig. 2.** Number of paths with exactly h-hops from source and destination

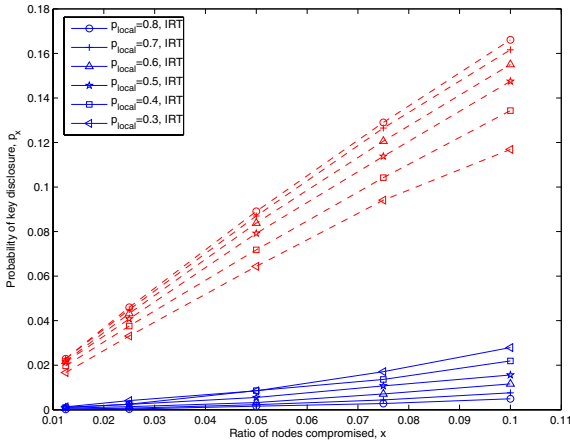**Fig. 3.** Average symbols that should be transmitted

Fig. 4. Secret disclosure probability of the IRT and SP schemes. The red dashed lines represent the performance of the SP scheme.

decreases. Note that there is a slight increase of number of transmitted symbols in the SP scheme as the ratio of nodes compromised, x, increases. This is due to the higher probability of the used path being compromised.

In Fig. 4, we show the secret disclosure probability of the IRT scheme and the SP scheme. The value of γ is set to 64. The curves with solid lines represent p_x of the IRT scheme for different local connectivity, p_{local}. The red dashed lines show the p_x of the SP scheme. We can observe a much higher secret disclosure probability of the SP scheme. Furthermore, p_x increases with x, as expected.

The flexibility of secret disclosure probability of the IRT scheme is presented in Fig. 5. In this figure, we vary the value of α and showed the probability of
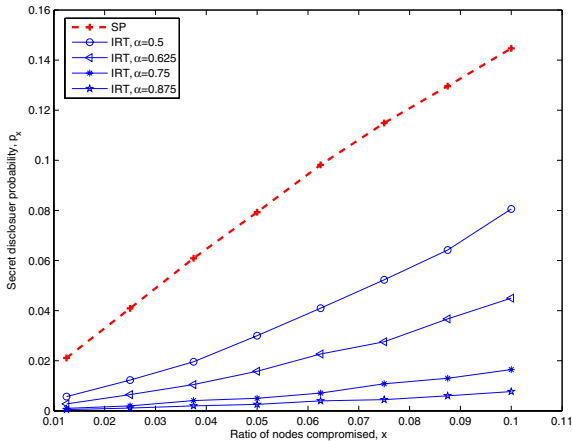
Fig. 5. The flexible secret disclosure probability of the IRT scheme

key disclosure for different node compromised ratio, x. It can be observed that the IRT scheme has a much lower p_x than the SP scheme when $\alpha < 1$. As α increases, the p_x value is smaller. Therefore, the IRT scheme provides a nice property of flexibility: a pre-defined threshold of probability of key disclosure can be guaranteed by varying (n, k).

7   Conclusions

We have proposed and investigated an Incremental Redundancy Transmission (IRT) scheme for the secret link key establishment process of key pre-distribution techniques. The IRT scheme uses the powerful MDS codes to encode the secret link key information and send through multiple multi-hop paths. The redundant symbols are transmitted only when they are needed to enable the destination to decode the secret information. One salient feature of the IRT scheme is the flexibility of trading transmission for lower information disclosure.

In the future work, we will consider sending different number of symbols along different paths based on the lengths of the paths in order to further reduce the overall transmission overhead and to improve the security performance. The effect of node disjoint paths will be investigated as well.

References

1. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proc. of the 9th ACM conference on Computer and communications security, Washington, DC, USA (2002) 41–47
2. Chan, H., Perrig, A., Song, D.: Random key predistribution schemes for sensor networks. In: Proc. of IEEE Symposium on Security and Privacy, Berkeley, California (2003) 197–213

3. Du, W., Deng, J., Han, Y.S., Varshney, P.K., Katz, J., Khalili, A.: A pairwise key pre-distribution scheme for wireless sensor networks. ACM Trans. on Information and System Security **8** (2005) 228–258
4. Liu, D., Ning, P.: Establishing pairwise keys in distributed sensor networks. In: Proc. of the 10th ACM Conference on Computer and Communications Security (CCS '03), Washington, DC, USA (2003) 52–61
5. Camtepe, S.A., Yener, B.: Combinatorial design of key distribution mechanisms for wireless sensor networks. In Samarati, P., Ryan, P., Gollmann, D., Molva, R., eds.: 9th European Symposium on Research Computer Security. Volume 3193 of Proceedings Series: Lecture Notes in Computer Science (LNCS)., Springer-Verlag (2004) 293–308
6. Lee, J., Stinson, D.R.: A combinatorial approach to key predistribution for distributed sensor networks. In: Proc. of IEEE Wireless Communications and Networking Conference (WCNC '05), New Orleans, LA, USA (2005)
7. Papadimitratos, P., Haas, Z.J.: Secure message transmission in mobile ad hoc networks. Elsevier Ad Hoc Networks **1** (2003) 193–209
8. Tsirigos, A., Haas, Z.J.: Multipath routing in the presence of frequent topological changes. IEEE Communication Magazine (2001) 132–138
9. Rabin, M.O.: Efficient dispersal of information for security, load balancing, and fault tolerance. Journal of the Association for Computing Machinery **36** (1989) 335–348
10. Pursley, M.B., Sandberg, S.D.: Incremental-redundancy transmission for meteorburst communications. IEEE Trans. on Communications **39** (1991) 689–702
11. Wicker, S.B., Bartz, M.J.: Type-II hybrid-ARQ protocols using punctured MDS codes. IEEE Trans. on Communications **42** (1994) 1431–1440
12. Du, W., Deng, J., Han, Y.S., Chen, S., Varshney, P.K.: A key management scheme for wireless sensor networks using deployment knowledge. In: Proc. of the 23rd Conference of the IEEE Communications Society (Infocom '04), Hong Kong, China (2004) 586–597
13. Wicker, S.B., Bhargava, V.K.: Reed-Solomon Codes and Their Applications. Piscataway, NJ: IEEE Press (1994)
14. Lou, W., Fang, Y.: A multipath routing approach for secure data delivery. In: Proc. of IEEE Military Communications Conference (MILCOM '01), McLean, VA, USA (2001) 1467–1473