# Enhancing Resilience of Probabilistic Key Pre-distribution Schemes for WSNs through Hash Chaining

Walid Bechkit
Universite de Technologie de Compiegne, Laboratoire Heudiasyc UMR CNRS 6599
Compiegne, France
wbechkit@hds.utc.fr

Abdelmadjid Bouabdallah
Universite de Technologie de Compiegne, Laboratoire Heudiasyc UMR CNRS 6599
Compiegne, France
bouabdal@hds.utc.fr

Yacine Challal
Universite de Technologie de Compiegne, Laboratoire Heudiasyc UMR CNRS 6599
Compiegne, France
ychallal@hds.utc.fr

## ABSTRACT

We propose, in this paper, a novel class of probabilistic key pre-distribution schemes highly resilient against node capture. We introduce a new approach to enhance resilience by concealing keys through the use of a simple hash chaining mechanism. We provide analytical analysis which shows that our solution enhances the network resilience against node capture without introducing a new overhead comparatively to similar solutions in the literature.

## Categories and Subject Descriptors

C.2.0 [**Computer-communication networks**]: General—*security and protection*

## General Terms

Design, Security

## Keywords

wireless sensor networks, security, key management, resilience.

## 1. INTRODUCTION

A wireless sensor network (WSN) is composed of a set of tiny autonomous sensor nodes with sensing, computation, and wireless communication capabilities. The purpose of such networks is to collect information issued from a controlled environment or a target object.

Wireless sensor networks are increasingly used in numerous fields such as military, medical and environmental sectors [1]. They are involved in a lot of critical applications, that's why security emerges as a crucial challenge in these networks. On the other hand, WSNs are very constrained which influences the security solutions.

Key management is a corner stone service for any security solution for WSNs. Asymmetric solutions are unsuitable because of resource limitations in WSNs. Pairwise secret key establishment is then one of the most suitable paradigm for securing exchanges in this kind of networks. Existing symmetric schemes can be classified into two categories: deterministic schemes and probabilistic ones:

Deterministic schemes ensure that each node is able to establish a pair-wise key with its neighbors. To guarantee determinism, protocols such as LEAP [2] make use of a common transitory key that is preloaded into all nodes prior to deployment. The transitory key is used to generate session keys between neighboring nodes and is cleared from the memory of nodes by the end of a short time interval after the deployment. The deterministic schemes are vulnerable to node compromise, if the transitory initial key is discovered the entire network can be compromised.

In probabilistic key pre-distribution schemes, each node is preloaded prior to deployment with a subset of keys taken from a large key-pool. Each two neighbor nodes have a certain probability of sharing a common key that belongs to both key subsets of those neighbors; the common keys are used to establish the secret session pairwise keys.

Eschenauer and Gligor proposed in [3] the basic Random Key Pre-distribution scheme that we denote by RKP. In this scheme, each node is pre-loaded with $m$ keys randomly selected from a large pool $S$ of keys. After deployment, each node exchanges messages containing its key identifiers with its neighbors. If two neighbors have at least one shared key they establish a secure link and compute their session secret key. Otherwise, they should find a secure path composed of secure links. This basic approach is CPU and energy efficient but it requires a large memory space to store the key ring at each sensor node. Also, if the network nodes are progressively corrupted, the attacker may discover a large part or the whole global key pool and hence, a great number of links will be compromised. Indeed, the same key may be used to secure different links in the network if it comes that this key is found common between key rings of different pair of nodes.

Chan et al. proposed in [4] a protocol that enhances the resilience of RKP. The main idea is that two neighbors 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 : $K_{i,j} = Hash(K_{s_1} \| K_{s_2} \| ... \| K_{s_{q'}})$ where $K_{s_1}, K_{s_2}, ...K_{s_{q'}}$ are the $q'$ shared keys between the nodes i and j. This approach enhances the resilience against node capture attacks because the attacker needs more overlap keys to break a secure link. We denote this protocol by Q-RKP for Q-composite Random Key Pre-distribution scheme where Q is the minimum required number of common keys to establish secure links.

*CCS'10,* October 4–8, 2010, Chicago, Illinois, USA.
ACM 978-1-4503-0244-9/10/10.

Castelluccia and Spognardi [5] adapted the RKP scheme to the multi-stage wireless sensor networks where new nodes are periodically deployed to ensure the network connectivity.

In this work, we introduce a new approach to enhance resilience of probabilistic key pre-distribution schemes by concealing keys through the use of a simple hash function mechanism. Compared to existing schemes, our solution enhances the network resilience against node capture without introducing any overhead. Indeed, the analytical study shows that our solution may reduce the fraction of compromised links up to 30%.

## 2. HC(X) : A NEW CLASS OF RANDOM KEY PRE-DISTRIBUTION SCHEMES

We define a new class of random key pre-distribution protocols that we denote by HC(x) for Hash-Chained(x) where x is an existing random key pre-distribution protocol. This new class of protocols enhances the resilience of existing random key pre-distribution schemes through a light hash chaining technique that conceals the same keys used to secure different links, as we will show in what follows.

Like in the existing probabilistic key pre-distribution schemes, we assume that a large pool $S$ of keys and their identifiers are generated off-line. Each node is preloaded with a key ring of $m$ keys randomly selected from $S$. Before the deployment, we propose to conceal keys in a way that the disclosure of some keys following a node capture reveals only derived versions that cannot be used to compromise links secured with the original keys.

The main idea of our new class consists of applying a hash function on the initial preloaded keys before deployment: a secure hash function $h$ is applied on each node keys $i \ mod \ N$ times where $i$ is the identifier of the node and $N$ is a preloaded value in all the network nodes. As known, the main characteristic of hash functions is that knowing a value of the chain it is computationally infeasible to determine the backward values. So, in our approach, when an attacker corrupts the key ring of a node $i$ it cannot discover keys having the same identifiers and used in a node $j$ such as $j \ mod \ N < i \ mod \ N$ and hence it compromises less links than in basic schemes.

This approach can be applied to any random key pre-distribution scheme. In what follows, we have chosen to apply it to Q-RKP with Q=1 to facilitate the explanation of the concepts behind our approach.

### 2.1 Application of our approach to 1-RKP

The application of our new approach to 1-RKP (Q-RKP with Q=1) [4] gives birth to a new resilient enhanced random key pre-distribution protocol HC(1-RKP).

The keys preloaded in a node $i$ are then:
$KR^i = \left\{h^{i \ mod \ N}(K_1^i), h^{i \ mod \ N}(K_2^i), ...h^{i \ mod \ N}(K_m^i)\right\}$
where $K_1^i, K_2^i, ...K_m^i$ are randomly selected from $S$.

As in 1-RKP, the pairwise secret key between two nodes in our solution is computed as the hash of all shared keys concatenated to each other.

Let us assume that the node $i$ shares $q'$ keys ($q' > 0$) with its neighbor $j$ and that $i \ mod \ N > j \ mod \ N$.
The node $i$ computes its secret key with $j$ as follows :

$$K_{ij} = Hash(K_{s_1}^i \| K_{s_2}^i \| ... \| K_{s_{q'}}^i)$$

where $K_{s_1}^i, K_{s_2}^i, ..., K_{s_{q'}}^i$ are the common keys with the node $j$.

The node $j$ computes its key as :

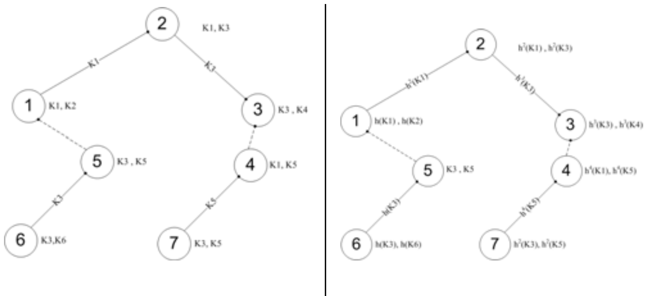
$$\begin{aligned} K_{ji} \quad &= \quad Hash(h^{i \ mod \ N - j \ mod \ N}(K_{s_1}^j) \| \\ & \qquad h^{i \ mod \ N - j \ mod \ N}(K_{s_2}^j) \| ... \| h^{i \ mod \ N - j \ mod \ N}(K_{s_{q'}}^j)) \end{aligned}$$

where $K_{s_1}^j, K_{s_2}^j, ..., K_{s_{q'}}^j$ are the common keys with the node $i$.

***Example:***

To illustrate our idea, let us refer to the figure 1. To simplify the comprehension, we assume that we have seven nodes and four secure links. When we use the 1-RKP scheme (Figure 1(a)), the corruption of the two nodes 4 and 7 induces the disclosure of the keys K1,K3 and K5 and then the compromise of the three external links (1,2),(2,3) and (5,6).

When we use our new protocol HC(1-RKP) (Figure 1(b)), keys are hashed before deployment (we assume in the example that N=5). The corruption of the two nodes 4 and 7 induces the disclosure of the derived keys $h^4$(K1), $h^4$(K5), $h^2$(K3) and $h^2$(K5). In this case, the link (2,3) can be compromised, however, the two other external links (1,2) and (5,6) cannot be compromised because it is infeasible to calculate $h^2$(k1) knowing $h^4$(K1) and $h$(k3) knowing $h^2$(K3).



(a) 1-RKP (b) HC(1-RKP)

**Figure 1: Example**

## 3. ANALYSIS

Let us calculate the fraction of compromised links when $x$ nodes are captured when we use the two schemes 1-RKP and HC(1-RKP). Note that we consider only links which are independent of the compromised nodes.

We summarize in table 1 the main symbols that we use in what follows:

**Table 1: SUMMARY OF NOTATIONS**

| | |
|---|---|
| $S$ | The global key pool |
| $\|S\|$ | The size of the global key pool |
| $m$ | The size of the node key ring |
| $P_{lc}(k)$ | The probability that two nodes share exactly k keys in their subset of keys. |
| $N$ | The protocol parameter |

## 3.1 Resilience analysis in 1-RKP

In 1-RKP , the fraction of discovered keys when a node is captured is $p = \frac{m}{|S|}$. The fraction of uncompromised keys is then $1 - p$. When $x$ nodes are compromised the fraction of uncompromised keys is $(1 - p)^x$. The probability that a given key has been known is then $1 - (1 - p)^x$. For any given link composed of $k$ shared keys the probability of being compromised is $(1 - (1 - p)^x)^k$.

Let $P_{lc}(k)$ be the probability that two nodes share exactly $k$ keys. By applying the law of total probability, it results that the probability of a link between two uncompromised nodes being compromised when $x$ nodes are compromised is:

$$P_{1-RKP} = \sum_{k=1}^{k=m} (1 - (1 - \frac{m}{|S|})^x)^k \frac{P_{lc}(k)}{\sum_{i=1}^{i=m} P_{lc}(i)} \qquad (1)$$

where $P_{lc}(k)$ can be found in [4].

## 3.2 Resilience analysis in HC(1-RKP)

In HC(1-RKP), the fraction of discovered keys when a node is captured is $p_h = \frac{N+1}{2N} \frac{m}{|S|}$. Let us assume that a key is discovered, this key is initially hashed $i$ times $(0 \leq i \leq N-1)$ with a probability $\frac{1}{N}$ . When the initial key is hashed $i$ times the probability that a key having the same identifier can be discovered is $\frac{N-i}{N}$. Thus, by applying the law of total probability we find that the probability to disclose a key having the same identifier with a compromised key is : $\sum_{i=0}^{i=N-1} \frac{1}{N} \frac{N-i}{N} = \frac{N+1}{2N}$

Since each node has $m$ keys, the probability that a key has been known when a node is compromised is so $p_h = \frac{N+1}{2N} \frac{m}{|S|}$. Following the same steps above we find:

$$P_{HC(1-RKP)} = \sum_{k=1}^{k=m} (1 - (1 - \frac{N+1}{2N} \frac{m}{|S|})^x)^k \frac{P_{lc}(k)}{\sum_{i=1}^{i=m} P_{lc}(i)} \qquad (2)$$

## 3.3 Comparison

From formulas 1 and 2, we notice that the probability of link compromise is less when we use the hash chaining mechanism that we propose in our solution. Thus our scheme is more resilient than the basic one.

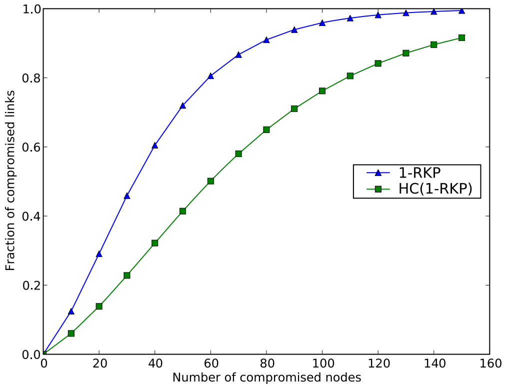


**Figure 2: Probability that an independant link be compromised when x nodes are captured**

Figure 2 plots the probability that an independent link be compromised depending on the number of captured nodes when we use the two scheme 1-RKP and HC(1-RKP). We have chosen $|S| = 1000$, $m = 40$ and $N = 10$ where N is the parameter of our solution. The figure shows clearly that the resilience against node capture attacks is better when we use our new approach. For instance, when the number of compromised nodes is between 40 and 80, our approach reduces the fraction of compromised links up to 30%.

## 3.4 Choice of N

We remark from proposition 2 that the higher is $N$ the better is the resilience. However when $N$ is too high the computation performance degrades because some nodes may apply hash function a high number of times to compute secret keys.

Since the limit of $N+1/2N$ in equation 2 as $N$ approaches infinity is $1/2$, we propose to take $N$ between 10 and 20. Hence the value $N + 1/2N$ is close to $1/2$. In addition, the computation performances are acceptable since nodes apply hash function a low number of times.

# 4. CONCLUSION

Asymmetric key management schemes are unsuitable because of resource limitations in WSNs. Pairwise secret key establishment is then the most suitable paradigm for securing exchanges in this kind of networks. In this paper we present a new class of key management schemes highly resilient against node capture. We enhance resilience by concealing keys through the use of a simple hash function mechanism without introducing any overhead. The analytical study shows that our approach may enhance resiliency up to 30%.

# 5. 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.

# 6. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38:393–422, 2002.

[2] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. Leap: efficient security mechanisms for large-scale distributed sensor networks. In *ACM CCS '03*, pages 62–72, New York, 2003.

[3] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *ACM CCS '02*, pages 41–47, New York, 2002.

[4] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In *IEEE SP '03*, Washington, 2003.

[5] C. Castelluccia and A. Spognardi. A Robust Key Pre-distribution Protocol for Multi-Phase Wireless Sensor Networks. In *IEEE Securecom'07*, 2007.