Securing Wireless Sensor Networks against Large-scale Node Capture Attacks

Tuan Manh Vu, Reihaneh Safavi-Naini and Carey Williamson
University of Calgary
Calgary, AB, Canada
{vuv,rei,carey}@ucalgary.ca

ABSTRACT

Securing wireless sensor networks against node capture is a challenging task. All well-known random key pre-distribution systems, including the Eschenauer and Gligor's pioneering scheme, its extensions, as well as threshold schemes, become insecure when a large number of nodes are captured. We propose a general technique, called *virtual key ring*, that can effectively strengthen the resilience of random key pre-distribution systems against node capture attacks by reducing the pre-loaded keying material while maintaining secure connectivity of the network.

The technique is general and applicable to many key pre-distribution systems. We however focus on the original EG scheme and propose a virtual key ring system based on this pioneering scheme. We provide detailed mathematical analysis and a security proof for the system, and use extensive simulation to validate the analysis and to compare performance of the new system with the original EG scheme. We also present simulation results for the strengthened resilience when the virtual key ring scheme is combined with the multi-path key reinforcement and q-composite techniques, showing that the system resilience is substantially improved against large-scale node capture attack (e.g., 40% of nodes captured).

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection (e.g., firewalls)*

General Terms

Design, Security

Keywords

key management, random key pre-distribution, sensor network security, resilience to node capture

ASIACCS'10 April 13–16, 2010, Beijing, China.

1. INTRODUCTION

Wireless sensor networks (WSNs) are *ad hoc* networks that can consist of hundreds or thousands of sensor nodes. Each sensor is a small battery-powered device with limited memory and computational capability, and a short-range wireless radio for communication. The sensors can measure physical and environmental conditions such as temperature, humidity, pressure, light, and sound. The sensed data is forwarded to a *base station* through multi-hop paths that are established in collaboration between sensor nodes.

To protect the transmitted information, pairs of sensors establish secret keys used for data encryption and decryption. The primary goal of a key establishment solution for WSNs is to minimize drain on the battery power while ensuring good connectivity and security. Eschenauer and Gligor [8] (EG) pioneered an innovative approach to the key establishment problem called *random key pre-distribution*. They suggest that every node is assigned a *key ring*, which is a random subset of keys of a *key pool*. Each key consists of a *key value* and a *key identifier*. Two nodes that are within the wireless communication range of each other can establish a secure channel, referred to as a *secure link*, if they possess at least one key in common. In order to find shared keys, nodes broadcast the key IDs in their key rings to wireless neighbors.

The EG scheme has guaranteed security against a *passive eavesdropping adversary* who only observes network communication. However, similar to other key pre-distribution approaches, the EG scheme is vulnerable to a *semi-honest adversary* who not only monitors the network traffic but also can capture sensor nodes. The semi-honest adversary obtains all keying material stored in the memory of captured nodes, and uses this information to decrypt the communication between uncaptured nodes. In the *semi-honest adversary model*, the captured nodes behave the same as others, making the attack devastating since it is undetectable by intrusion detection systems. This type of adversary is also known as the node capture adversary.

Eschenauer and Gligor considered the effect of node capture attack and showed that the resilience of their scheme deteriorates (almost linearly for small scale node capture) with the number of captured nodes. There have been several approaches [5, 7, 13] aimed at mitigating the consequence of this attack. Two notable solutions are *multi-path key reinforcement* and *q-composite*, which were developed by Chan *et al.* [5] to improve resilience of the EG scheme given the same network model (i.e., random deployment, homogeneous sensors, and so on).

The multi-path key reinforcement scheme uses the premise that two nodes can negotiate a new secure key (i.e., this key cannot be recovered by the adversary) if there exists at least one non-compromised path between them. In particular, for c captured nodes, the existence of $c+1$ disjoint paths guarantees a secure key can be established. The q-composite scheme improves resilience of the EG scheme by requiring more common keys for the establishment of a secure link: two nodes need to have at least q ($q \geq 2$) keys in common to be able to establish a secure link. Improved resilience using these methods completely breaks down for large-scale node capture attacks: the number of required paths between nodes becomes very large and q-composite schemes as noted in [5] will lose its effectiveness.

The objective of this paper is to address this problem, that is providing high resilience against *large-scale node capture*. We emphasize that capturing a node results in its keying material being accessible to the adversary. The node, however, functions the same as an uncaptured node. In other words, captured nodes follow the prescribed protocol; thus, the network remains connected and functional. In this way the adversary will remain undetected while attacking the network (i.e., capturing more nodes and learning the communication between uncaptured nodes as the attack goes on).

Our contribution

We introduce the concept of *virtual key rings* for key pre-distribution schemes to strengthen security of existing random key pre-distribution systems against node capture, and in particular against large-scale attack.

The virtual key ring of a node consists of two kinds of elements: a set of keys that includes the key values and the corresponding key identifiers, and an additional set of key identifiers of keys that belong to the first hop neighbors of the node that have secure links to the node. For a key identifier k_{ID} in the node's virtual key ring, the node can obtain strings that are computed using the corresponding key value k either i) by itself if it possesses k, or ii) from the one-hop neighbor node that has the key and has a secure link with the node.

The virtual key ring concept allows nodes to start with smaller initial key rings but enjoy the benefits of larger key rings through their virtual key rings. Larger effective key rings result in higher levels of secure connectivity and better resilience against node capture, especially against large-scale node capture attacks. To demonstrate the application of this concept in a concrete way, we present a virtual key ring key pre-distribution system based on the original scheme of EG, and provide detailed analysis of the scheme. We show that the new system has the same level of security as the EG scheme against passive adversary, and has improved security against node capture. The superior performance of the scheme against large scale node capture is best demonstrated when it is used in conjunction with the multi-path key reinforcement technique and the q-composite scheme, as shown below.

A key pre-distribution scheme using virtual key rings

Applying the virtual key ring concept to the EG scheme results in a new system consisting of a key pre-distribution phase and a two-round key establishment phase. In the key pre-distribution phase, nodes receive an *initial key ring* that is a subset of keys chosen randomly from a key pool. These keys are referred to as *primary keys*. Key establishment has two rounds: in the *first round*, nodes use a shared key discovery protocol similar to the EG scheme, and establish secure links with wireless neighbors with whom they have at least one key in common in their primary key rings. Two wireless neighbors that can establish a secure link in this round are *first-round trust neighbors* of each other. Each node also constructs a *virtual key ring* that consists of (i) its primary keys, and (ii) the *key IDs* of the keys in the initial key rings of its first-round trust neighbors. Virtual key rings enable the nodes to establish unique *pairwise keys* with many of their wireless neighbors that are not first-round trust neighbors. Pairwise key establishment in the second round uses an efficient protocol with provable security. Secure connectivity for the network is achieved using links established in both rounds. The two-round key establishment results in the majority of secure links being protected by unique pairwise keys. Additionally, each node is pre-loaded with fewer keys compared to the EG scheme for the same connectivity. In other words, the adversary learns less information from each captured node. For these reasons, the virtual key ring concept strengthens the resilience against node capture attacks.

We give a complete security and efficiency analysis of the protocol. The security analysis consists of (i) proving security of the key establishment protocol, and (ii) calculating the link compromise probability when c nodes are captured. Step (i) shows that the security of the virtual key ring scheme against a passive eavesdropping adversary is the same as the security of the EG scheme (i.e., the additional round of key establishment with virtual key rings does not weaken the system resilience against the passive adversary). Besides, the probability analysis suggests strengthened resilience against a semi-honest adversary, compared to the EG scheme. The efficiency analysis includes estimating initial key ring size to ensure high connectivity, and examining the extra computation as well as communication cost.

Further strengthening of the resilience

The multi-path key reinforcement and the q-composite techniques are the two main solutions aimed at improving resilience of the EG scheme against node capture. These techniques lose their effectiveness when a large number of nodes are captured.

The virtual key ring concept allows nodes to have much larger effective key rings (compared to their initial key rings). The additional round of key establishment with virtual key rings results in many more secure links in the network and thus substantially increases the number of disjoint paths between two nodes. This extends the effectiveness of the multi-path key reinforcement technique against a much larger number of captured nodes.

Using virtual key rings also boosts effectiveness of the q-composite technique. Larger effective key rings (i.e., virtual key rings) mean that the number of shared keys between two nodes can greatly increase and so the link key can potentially rely on many more keys. This means that it would be much harder for the adversary to reconstruct a link key that is computed as a function of a large number of shared keys between the two virtual key rings. These arguments are verified by our simulation results and summarized in Figure 3.

Our analysis results show that the EG scheme with and without the multi-path key reinforcement technique, and the q-composite scheme, provide minimal security for un-

captured nodes when a large number of nodes are captured. For example, when half of the nodes are captured, the survival probability of a secure link between any two uncaptured nodes is less than 10%. (The *link survival probability* is the probability that a link is *not compromised*.) Under the same scenario (i.e., half of the nodes are captured), the link survival probability of our proposed scheme when combined with the q-composite scheme is greater than 70%.

Proofs and simulations
In our analysis, we use security proofs, analytical derivations, and extensive simulation studies. Introducing virtual key rings raises a difficulty in estimating the initial key ring size for achieving secure connectivity. This is because the virtual key rings are a function of nodes' neighbors and so the size of the virtual key ring is different in different nodes. This means that, unlike the EG scheme, it is not possible to use Erdős-Rényi's random graph theory [15] to model secure connectivity of the network. To find the size of initial key rings, we rely on extensive simulation results to estimate the initial key ring size that is required for achieving secure connectivity in the network with high probability (i.e., 99.9%). We derive the probabilities of link compromise for the proposed virtual key ring scheme, and use simulations to verify the mathematical analysis. We evaluate performance of the schemes obtained by combining the virtual key ring scheme with the multi-path key reinforcement and q-composite techniques, through extensive simulations.

To perform systematic simulations we have developed an efficient simulation framework for studying random key predistributions for WSNs (i.e., up to 50,000 nodes). The simulation of WSNs can be performed for various sizes, network densities, and key pre-distribution parameters (i.e., the key pool size and the initial key ring size). In each case, the secure connectivity of the network is determined and the minimum size of the key ring that achieves connectivity is found. Using this tool and extensive simulations, we obtain a heuristic expression for estimating, with a high level of accuracy, the initial key ring size as a function of network density and the key pool size. The details of the simulator, including the software components, the functionalities, and the download URL, are presented in [17].

The rest of this paper is organized as follows. In Section 2, we review related work. In Section 3, we state the system assumptions, introduce the notation used in the paper, and briefly describe the EG and q-composite schemes. We present a new key pre-distribution scheme obtained by applying the virtual key ring concept to the EG scheme in Section 4, and analyze its performance and security in Section 5. Section 6 presents simulation results, and Section 7 concludes the paper. Appendices A to C provide formal proofs for our theoretical results.

2. RELATED WORK

There have been numerous efforts to increase the resilience of the EG scheme against node capture. Chan *et al.* [5] proposed three methods, two of which use the same key assignment approach as the EG scheme, while the other uses a different approach. In the first scheme, called the *q-composite scheme*, two wireless neighbors will set up a secure link if they share at least q keys. They showed that increasing q will improve resilience against *small-scale attack* but as the number of captured nodes grows, the system's resilience deteriorates and becomes worse than the EG scheme. The second technique, called *multi-path key reinforcement*, improves security against node capture by using multiple paths between two nodes to send portions of a new key. The technique, however, has high computation cost for finding disjoint paths and high communication cost for sending messages. Chan *et al.* also proposed a scheme that provides high resilience against node capture by manually assigning unique keys to selected pairs of nodes. This drastically reduces flexibility compared to EG and q-composite schemes and makes it unsuitable for the scenarios where the network size is not known beforehand.

Other approaches to key distribution in WSNs include using deterministic algorithms [1, 2] to assign key rings to nodes using special combinatorial designs, location-aware schemes [6, 12, 9, 11] that assume a specific WSN topology, and threshold schemes [7, 13] that are designed to provide strong security when the number of captured nodes is below a certain threshold. Limitations of these schemes are the rigid choice of system parameters in combinatorial schemes, limited flexibility in deployment for location-aware schemes, and complete loss of security when the number of captured nodes exceeds the design threshold.

3. PRELIMINARIES

We assume that *the sensors are distributed randomly and uniformly within a planar square region*. Additionally, *sensors all have the same communication range, and there are no obstacles between them*. The network size and density may vary depending on the applications. We consider WSNs with 1,000 or more sensors, and assume the density is 30 or higher. These numbers are chosen experimentally to ensure wireless connectivity, which is a pre-requisite for secure connectivity.

3.1 Notation and terminology

Table 1 summarizes the notation used throughout the paper. Two nodes within the communication range of each other are *wireless neighbors*. Two wireless neighbors that are able to compute a shared secret and thus can establish a secure communication channel are *trust neighbors*. The secure communication channel between two trust neighbors is a *secure link*. A *secure path* is a sequence of adjacent secure links. A deployed network is said to be *securely connected* if there exists a secure path between every pair of nodes. Sensor nodes are scattered randomly, and keys are pre-distributed randomly. This may result in the deployed network not being connected. Random key pre-distribution schemes need to be designed such that the deployed network is securely connected with *high probability*.

An adversary can *capture* a node. Subsequently, the adversary can access all the information stored in the captured node, allowing the decryption of information transmitted on other formerly-secure links in other parts of the network. A link is compromised if its associated key is (i) a primary key that belongs to a node captured by the adversary, or (ii) a secondary key that can be reconstructed by the adversary.

Table 1: Notation Used for Mathematical Analysis

N	Size of the key pool
m	Size of an initial key ring
m_v	Size of a virtual key ring
n	Number of sensor nodes in the network
n_w	Average number of wireless neighbors for a sensor node (density)
c	Desired connectivity probability for WSN
p_i	Probability that two wireless neighbors can set up a secure link in round i
n_{t_i}	The expected number of trust neighbors that a node can discover in round i
n_t	The expected total number of trust neighbors for each node (node degree)
x	Number of captured sensor nodes

3.2 EG and q-composite schemes

EG scheme:
In the EG scheme, each sensor has a randomly selected key ring of size m, drawn uniformly at random from a *key pool* of N keys. Once deployed, two sensor nodes use the *shared-key discovery phase* to establish secure links with their wireless neighbors with which they share a key.

Using Erdős-Rényi's random graph theory [15], one can estimate m for a given N. A random graph $G(n,p)$ is a graph of n nodes in which each of the $\frac{n\cdot(n-1)}{2}$ possible edges occurs independently with probability p. It has been shown that as n approaches infinity, a random graph $G(n,p)$ is connected with probability c if the edge probability $p = \frac{1}{n}\cdot(\ln(n) - \ln(-\ln(c)))$. This leads to the expected node degree $n_t = (n-1)\cdot p = \frac{n-1}{n}\cdot(\ln(n) - \ln(-\ln(c)))$ if the network size is n and the desired connectivity is c.

On the other hand, let $\bar{p}$ be the probability that two randomly chosen key rings share at least one key. Due to the wireless connectivity constraints, a sensor node can only set up secure links with its wireless neighbors. Thus, the expected number of trust neighbors of a sensor node is $n_t = n_w \cdot \bar{p}$. Given a desired node degree n_t and an average density n_w, the ring size m and pool size N must be chosen such that $\bar{p} \geq \frac{n_t}{n_w}$. In particular, Eschenauer and Gligor showed that $\bar{p} = 1 - \frac{((N-m)!)^2}{N!\cdot(N-2\cdot m)!}$. Either N or m can be set to a fixed value, and the other parameter can be computed accordingly.

q-composite scheme:
The q-composite approach is an extension of the EG scheme, in which two wireless neighbors can establish a secure link when they have at least $q \geq 2$ common keys. The hash value of the concatenation of all common keys becomes the pairwise secret key.

In the q-composite scheme, the probability that two arbitrary rings share at least q keys is $\bar{p} = \sum_{i=q}^{m} s(i)$, where $s(i)$ is the probability two arbitrary rings have exactly i common keys. Specifically, $s(i) = \frac{((N-m)!)^2\cdot(m!)^2}{N!\cdot(N-2\cdot m+i)!\cdot i!\cdot((m-i)!)^2}$. Similar to the EG scheme analysis, given a network size n, an average density n_w, a desired connectivity c, and a security parameter q, if one of the two parameters N and m is fixed, the other can be computed such that $\bar{p} \geq \frac{n_t}{n_w} = \frac{1}{n_w}\cdot\frac{n-1}{n}\cdot(\ln(n) - \ln(-\ln(c)))$.

4. EG WITH VIRTUAL KEY RING SCHEME

4.1 Description

Applying the virtual key ring technique to the original EG system results in a new scheme that consists of an off-line key pre-distribution phase followed by *two rounds* of key establishment. The off-line phase and the first round of key establishment are the same as in the EG scheme. Specifically, every sensor is loaded with a key ring, which is a set of m distinct keys chosen uniformly at random from a key pool of size N. These keys are referred to as *primary keys*. Additionally, all sensors share a pseudo-random function $F(k,X)$ where k is a primary key and X is an input string. Examples of $F(k,X)$ are secure encryption or message authentication functions. Details are provided in Section 4.4.

First-round key establishment:
During the sensor network initialization, each sensor node advertises the identifiers of its primary keys. Any pair of wireless neighbors sharing a primary key will set up a secure link. The trust neighbors discovered in this round are called *first-round trust neighbors*.

Each sensor node also maintains a list of the identifiers of primary keys that belong to its first-round trust neighbors. The *virtual key ring* of a node consists of (i) its primary keys (and their identifiers) and (ii) the identifiers of primary keys that belong to its first-round trust neighbors.

Second-round key establishment:
In the second round, each sensor node broadcasts the identifiers of the keys in its virtual key ring to find more trust neighbors. Two wireless neighbors that cannot form a secure link in the first round but have at least one common key in their virtual key rings can obtain a shared secret key following the protocol in Table 2. Note that for two wireless neighbors who share a key ID in their virtual key rings, but cannot establish a secure link in the first round, it must be the case that at most one of the nodes has the actual key.

Assume that A and B form such pair. Without loss of generality, let A start the protocol. A picks one of the keys in its virtual key ring that is shared with B, say k. A generates a random value r for X (according to the specification of $F(k,X)$), and sends $KeyID_k$ (the identifier of key k) and the random string r to B. Once B receives $KeyID_k$ and r, both A and B can obtain $F(k,r)$ as described in the following protocol. $F(k,r)$ will be the shared secret key that A and B will use to establish a secure link between them.

Table 2: The protocol establishes a secure key, and hence a secure link, between nodes A and B in the second round. Communications that are labeled 'secure' are sent over secure links, encrypted with a primary key.

$A \longrightarrow B$	:	$KeyID_k \| r$
If A does not have the key k		
$A \longrightarrow A'$	:	$KeyID_k \| r$
$A' \longrightarrow A$ (securely)	:	$F(k,r)$
If B does not have the key k		
$B \longrightarrow B'$	:	$KeyID_k \| r$
$B' \longrightarrow B$ (securely)	:	$F(k,r)$

Since k is in A's virtual key ring, k belongs to either A or at least one of its first-round trust neighbors, say A'. If A indeed possesses k, A can directly compute $F(k, r)$. Otherwise, A sends a message of the form $(KeyID_k, r)$ to A' and asks A' to compute $F(k, r)$. Upon receiving the request, A' computes $F(k, r)$ and transfers the result *over a secure link* back to A (since A and A' are first-round trust neighbors, they have a secure link). Similarly, B calculates $F(k, r)$ itself or securely obtains it from B', one of its first-round trust neighbors that has k. At the end of the protocol, both sensor nodes A and B share a new secret key that can be used to protect their future communications.

We assume that *at the end of the key exchange protocol, only A and B keep $F(k, r)$, and nodes that assist with the computation of $F(k, r)$ (i.e., A' and B') erase that from their memories*. This is an important assumption that is used in arguing security of the system.

Discussion: One may argue that the second round of key establishment is similar to other approaches in which A generates and sends a pairwise key to B via one or more multi-hop secure paths such as $A \rightarrow A' \rightarrow B' \rightarrow B$. However, the key establishment in the second round only requires A and B to be wireless neighbors, and it works *regardless of the secure connectivity between A and B after the first round*. For example, A' and B' could be out of communication range of each other and thus the path $A \rightarrow A' \rightarrow B' \rightarrow B$ does not exist. Also, if there exists no secure path from A to B at all, the proposed protocol still works. The example in Section 4.2 illustrates this claim more clearly.

Even if in the scenario that A and B do have one or more secure paths to negotiate a pairwise key, finding paths and transferring key shares along multi-hop path(s) in WSNs are very costly in terms of communication overhead and implementation efforts. The way that the nodes perform the second round of key establishment with virtual key rings is only involved with the communication between one-hop neighbors.

In short, the second round of key establishment provides a simple mechanism for nodes to establish additional secure links regardless of their secure connectivity. The protocol is also applicable to other key pre-distribution systems, not just the EG scheme. The main concern is the security of second round key establishment, which is discussed in Section 4.3.

4.2 Example

The following example illustrates how the virtual key ring technique works with the EG scheme. Assume that in a network there are five sensor nodes whose wireless connectivity is depicted in Figure 1a. Each sensor possesses two random keys from a pool of five keys. Figure 1b shows the keys of each sensor and secure links formed in the first round of key establishment.

Figure 1c illustrates the virtual key rings of all sensor nodes and the additional secure links established in the second round. Solid lines represent first-round links, while dashed lines show second-round links. Since sensor node E cannot discover any trust neighbors in the first round, its virtual key ring is exactly the same as its initial key ring. In contrast, the virtual key rings of the other four sensor nodes have extra keys. For example, sensor node D has three additional keys (1, 3, and 5) in the virtual key ring

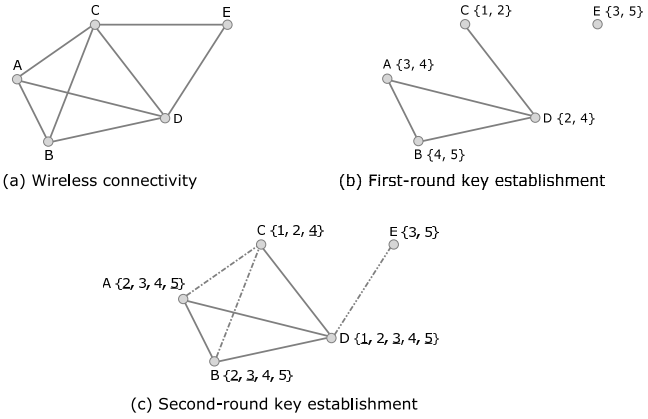


Figure 1: Example of two-round key establishment with virtual key rings.

from first-round trust neighbors C, A, and B, respectively. Node D does not actually possess these three keys, but it knows which trust neighbor has each of them.

With the additional keys, node D can set up a secure link with node E in the second round, despite the fact that node E's virtual key ring has not grown compared to its initial key ring. Assuming that node D chooses key 3 as the generation key, node E computes the pair-wise secret key itself, while node D obtains it from node A, with which it has already established a secure link.

In this example, sensor node E was still isolated from the other nodes after the first round of key establishment. Fortunately, the additional secure links formed in the second round make the network connected. In addition, they may reduce the path length between some already-connected nodes. As can be seen from Figure 1b, after the first-round key establishment, node B has to rely on node D in order to exchange information securely with node C. Once all nodes have established additional links with the virtual key rings, nodes B and C can communicate directly over their own secure channel. The second round of key establishment helps reduce the transmission cost between nodes A and C as well.

In summary, any two wireless neighbors that have at least one common key in their virtual key rings (but not in their initial key rings) can establish a secure link in the second round. The advantage of using the virtual key ring approach is that *better connectivity can be achieved with fewer initial keys* compared to the EG scheme, using one (or more) additional round(s) of key establishment. The primary benefits of the smaller initial key ring are *greater resilience* to node capture, and *lower memory requirements* for sensor nodes. The drawback is *higher communication cost for network initialization*. These issues are analyzed mathematically in Section 5.

4.3 Adversary model

We consider a *semi-honest* adversary whose aim is to learn the communication in the network, but follows the protocol in order to remain undetected. We assume the adversary can capture sensor nodes and access the information stored in them. We consider two cases (i.e., Late and Early) depending on the time of arrival in the system, and refer to them as adversaries $\mathcal{A}_L$ and $\mathcal{A}_E$.

Late adversary
$\mathcal{A}_L$ attacks a deployed system after completion of the initialization and key establishment. It randomly captures a subset of nodes. This adversary attacks the system during its normal operation. We consider this to be the more prevalent type of adversary, since node deployment is supervised in most applications.

Early adversary
$\mathcal{A}_E$ is an all-powerful adversary who is present from the start and can capture nodes and monitor all the network traffic at any time. We assume in both cases that the adversary selects nodes to capture uniformly at random. Note that the main difference between $\mathcal{A}_L$ and $\mathcal{A}_E$ is that having access to nodes as well as the messages exchanged during the second-round key establishment allows the adversary $\mathcal{A}_E$ not only to learn the primary keys stored in the nodes but also to reconstruct secondary keys (given enough information).

4.4 The basic protocol and its security

Consider the following protocol

Protocol $\mathbf{\Pi}$:		
$A \longrightarrow A'$	:	$KeyID_k \| r$
$A' \longrightarrow A$ (securely)	:	$E(k_{AA'}, F(k,r))$

where $E(.,.)$ is an *IND-CPA secure symmetric key encryption function* and $F(.,.)$ is a *length-preserving keyed pseudo-random function*. Let $k^* = F(k,r)$ denote the secondary key received by A as a result of the execution of protocol $\mathbf{\Pi}$.

An encryption function provides IND-CPA security if the ciphertext of a given challenge plaintext is indistinguishable from a random string of the same length, for an adversary who has been able to query the encryption oracle on polynomially many inputs, other than the challenge plaintext.

A *length-preserving keyed function* is a two-parameter function $F(k,X)$ where the first parameter is a *key* and the second parameter is simply called the input. The function is length-preserving if the key, input, and output all have the same length. A length-preserving keyed function is *pseudo-random* if for a fixed unknown value of key, knowing polynomially many input and output pairs gives negligible advantage in distinguishing the function from a random function.

We have the following theorem that shows that the secondary key k^* is a secure key.

THEOREM 1. *Let $E(.,.)$ be an IND-CPA symmetric key encryption function and $F(.,.)$ be an efficient length-preserving keyed function that is pseudo-random. Consider a pair of nodes A and A' that share a primary key $k_{AA'}$. Assume A knows the ID_k of a primary key k that is in the key ring of A'. Then $k^* = F(k,r)$ obtained by A through the execution of the protocol $\mathbf{\Pi}$ (received over the secure channel as $E(k_{AA'}, k^*)$) has the following properties.*

- *P1: A passive adversary who is present from the start of the deployment cannot learn anything about k^*.*
- *P2: An adversary $\mathcal{A}_L$ that has captured several nodes cannot learn anything about k^*. This is true even if $\mathcal{A}_L$ learns the primary key k that is used in $k^* = F(k,r)$.*
- *P3: An adversary $\mathcal{A}_E$ whose set of captured nodes does not include the primary key k (used for the key generation) and the primary key $k_{AA'}$ (used for securing the key transfer) does not learn anything about k^*.*

See Appendix A for the security argument for protocol $\mathbf{\Pi}$. The above theorem guarantees that the passive adversary does not learn anything about secondary keys. Therefore, the virtual key rings have the same security level against the passive adversary as key rings in the EG scheme. Adversary $\mathcal{A}_L$ can only learn the secondary keys that it finds in a set of captured nodes C. We note that the effect of $\mathcal{A}_L$ on compromising the key rings of nodes that are not in C is smaller than in EG, because for the same connectivity level fewer primary keys are used in the virtual key ring approach, and secondary keys are shared by only one other node (a trust neighbor of some captured node in C). An adversary $\mathcal{A}_E$ that captures a set C of nodes learns the secondary keys in the virtual key rings of nodes in C, and also all secondary keys that it can calculate using the primary keys obtained from C. This means that capturing the nodes in C will affect a larger set of virtual key rings and not only the secondary keys of the trust neighbors C. However, it can be shown that on average the effect on the compromise of other virtual key rings is less than the compromise of key rings in EG. These results are confirmed with simulations (Section 6.3) that compare the effect of node capture in different schemes.

5. MATHEMATICAL ANALYSIS

In this section, we study the secure connectivity, investigate the strength of the proposed scheme against different adversaries, and examine the protocol efficiency.

5.1 Secure connectivity

We first estimate the size of a virtual key ring, then calculate the probability that two wireless neighbors can establish a secure link in the second round. Using these results enables the analysis of the system resilience against node capture as well as the protocol efficiency.

5.1.1 Expected virtual key ring size

LEMMA 1. *For network density n_w, key pool size N, and initial key ring size m, the expected virtual key ring size is:*

$$m_v = m + \left[1 - \left(1 - \frac{\sum_{i=1}^{m}(g(i)\cdot(m-i))}{(N-m)\cdot\sum_{i=1}^{N} g(i)}\right)^{n_{t_1}}\right]\cdot(N-m) \quad (1)$$

where $n_{t_1} = \left(1 - \frac{((N-m)!)^2}{N!\cdot(N-2\cdot m)!}\right)\cdot n_w$ is the expected number of first-round trust neighbors, and $g(i) = \frac{(N-m)!\cdot m!}{(N-2\cdot m+i)!\cdot i!\cdot((N-i)!)^2}$.

A formal proof of this result appears in [16].

5.1.2 Probability of secure link establishment in round two

Secure links can be established in round one or two. In round one, the probability of link establishment is the same as EG scheme and is determined as $p_1 = 1 - \frac{((N-m)!)^2}{N!\cdot(N-2\cdot m)!}$ [8]. The following lemma estimates the probability of link establishment in the second round.

LEMMA 2. *Given the key pool size N, initial key ring size m, and virtual key ring size m_v, the probability of establishing a new secure link between two wireless neighbors in the second round is:*

$$p_2 = 1 - \frac{(N-2\cdot m)!\cdot((N-m_v)!)^2}{(N-2\cdot m_v)!\cdot((N-m)!)^2} \quad (2)$$

A formal proof of this result appears in [16].

Discussion: On average, each sensor node has $(n_w - n_{t_1})$ wireless neighbors with which it cannot set up secure links in the first round. Hence, it is expected that a sensor node can add

$$n_{t_2} = p_2 \cdot (n_w \ - \ n_{t_1}) \tag{3}$$

trust neighbors in the second round, bringing the total number of trust neighbors to $n_t = n_{t_1} + n_{t_2}$.

Equation 2 estimates the expected probability that a sensor node can set up secure links in the second round; nevertheless, this probability in fact varies a lot. In particular, the more first-round trust neighbors a sensor node has, the larger its virtual key ring size is. A larger virtual key ring increases the chance that this sensor node can establish secure links in the second round. Since the probability that any two wireless neighbors can set up a secure link in the second round is not fixed, the graph representing the WSN *loses its randomness properties*, making it inappropriate to use random graph theory in the study of network connectivity. The virtual key ring is designed so that the WSN is securely connected at least 99.9% of the time. Section 5.3 explains how the initial key ring size m is estimated to achieve such connectivity.

5.2 Resilience against node capture for virtual key ring scheme

We evaluate the resilience against node capture by (i) constructing a graph that consists of all the uncaptured nodes together with the secure links among them, and (ii) finding the fraction of links in this graph whose associated keys are known to the adversary via data from captured nodes. This fraction is denoted by $p_{c_L}(x)$ and $p_{c_E}(x)$, for the adversary $\mathcal{A}_L$ and $\mathcal{A}_E$ (as defined in Section 4.3), respectively. We have the following results.

LEMMA 3. *Consider an adversary that can capture x sensor nodes. The probability of compromise of a link for the two types of adversaries is given by the following probabilities:*

- *For adversary $\mathcal{A}_L$,*

$$p_{c_L}(x) = \frac{n_{t_1}}{n_{t_1} + n_{t_2}} \cdot \left(1 - (1 - \frac{m}{N})^x\right) \tag{4}$$

- *For adversary $\mathcal{A}_E$,*

$$\begin{aligned} p_{c_E}(x) \ &= \ p_{c_L}(x) + \hat{t} \cdot \left(1 - (1 - \frac{m_v - m}{m_v} \cdot \frac{x}{n})^2\right) \\ &\quad + \hat{t} \cdot (1 - \frac{m_v - m}{m_v} \cdot \frac{x}{n})^2 \cdot \left(1 - (1 - \frac{m}{N})^x\right) \end{aligned} \tag{5}$$

where n_{t_1} and n_{t_2} are the expected number of first-round and second-round trust neighbors, m_v is the expected size of the virtual key ring, and $\hat{t}$ is the fraction of second-round trust neighbors which is determined by $\frac{n_{t_2}}{n_{t_1}+n_{t_2}}$.

Proofs of Equations 4 and 5 are presented in Appendix B.

Discussion: The adversary A_L is weaker than the adversary A_E since A_L attacks the network after deployment. That means that the adversary A_L cannot reconstruct any secondary pairwise key k' between two uncaptured nodes as the input random string r required for computing k' is unknown to A_L. It is expected that when most of the nodes are captured, the adversary A_L can compromise all first-round secure links but all second-round links between uncaptured nodes remain secure. Hence, this leads to a great resilience of a virtual key ring based approach against A_L. However, this property can be easily achieved in the EG scheme and other systems as well, if two wireless neighbors use their pairwise key to negotiate a session key, and use the session key instead of using the pairwise key directly. For this reason, in Section 5.5, we are only interested in the comparison of the resilience against the stronger adversary A_E.

5.3 Memory storage

In the proposed scheme, the main memory requirement includes the initial key ring, virtual key ring, a random generator, and an IND-CPA cryptographic function. Among these, only the initial key ring is significant. This section discusses how the initial key ring size m is estimated to achieve high connectivity.

Since the random graph model is not applicable for analyzing the secure connectivity in the EG with virtual key ring scheme, we use simulation results to derive an equation for estimating the initial ring size. Let n_w denote the network density of a network of size at least 1,000 nodes that are deployed randomly in a planar square region. For a key pool of size N, the following expression gives an estimation $\hat{m}$ for the initial key ring size that achieves a connectivity of at least 99.9%:

$$\hat{m} = \left(\frac{2}{3}\right)^{\frac{n_w}{30}} \cdot \sqrt{N} \tag{6}$$

More details on the development of this expression are given in Appendix C.

5.4 Computation cost

Compared to the EG and q-composite schemes, using virtual key rings introduces extra computation and communication during the key establishment process. The computation cost involves generating a pseudo-random number r, and evaluating a pseudo-random function. Both are very efficient and thus the computation cost can be neglected. The main additional cost is the communication overhead between nodes during the second round of key establishment, which is examined in the next section.

5.5 Communication cost

The extra communication overhead occurs only during WSN initialization, which is a small portion of the total operational lifetime for the network. To simplify the calculation, we focus on the main transmissions during the key agreement process. Assume that a key is L_1 bits and a key ID is L_2 bits. If the key pool consists of N keys, $L_2 = \lceil \log_2(N) \rceil$ bits.

LEMMA 4. *The expected communication overhead in bits per sensor node is:*

$$(m+m_v)\cdot L_2 + \frac{n_{t_2}}{2}\cdot(L_1+L_2) + \frac{m_v - m}{m_v}\cdot n_{t_2}\cdot(2\cdot L_1 + L_2) \tag{7}$$

A formal proof of this simple counting problem appears in [16].

6. SIMULATION RESULTS

We use simulation (i) to study network connectivity, and (ii) to compare the resilience between the known key pre-distribution systems and their strengthened versions with virtual key ring technique. For space reasons, the validation of our theoretical analysis is not included in this paper, but is available in [16].

6.1 Initial key ring for high connectivity

Equation 6 approximates the initial key ring size to achieve a connectivity of at least 99.9% for a network of 1,000 or more sensor nodes that are deployed randomly in a planar square region. Our first set of simulation experiments studies the accuracy of this initial key ring size estimation. In order to do so, we select a number of sample key pool sizes distributed evenly over the range from 1,000 to 100,000. Also, the network size is 1,000 and the density varies from 30 to 60. For each set of parameters (n, n_w, N), we do binary search for the smallest initial key ring size such that the network is securely connected at least 9,990 times out of 10,000 simulation runs. The simulation results and our estimation of initial key ring size are plotted in Figure 2.

The comparison suggests that the estimation is very close to the simulation results. According to our statistics, when the key pool size is less than 30,000, the difference between the estimation and simulation results of initial key ring size is less than 7%. This difference is less than 10% for larger key pool sizes, up to 100,000.

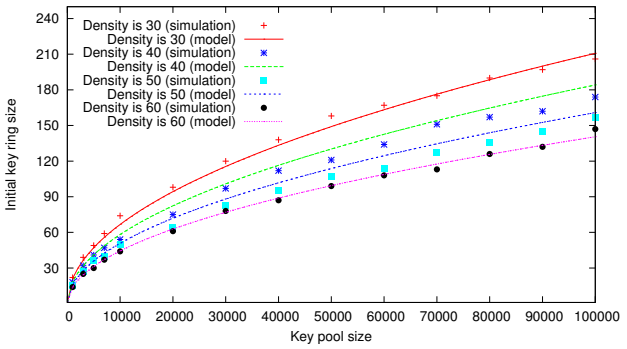


Figure 2: Comparison between the estimation and the simulation results.

6.2 Network connectivity

Recall that we consider a network to be highly connected if it is connected at least 9,990 times out of 10,000 simulations (i.e., the connectivity is roughly 99.9%).

We simulate a network of 1,000 sensor nodes that are scattered randomly and uniformly within a unit square. The communication range of the sensors is defined so that the average number of wireless neighbors of a sensor node is 30. The key pool is set to 50,000 keys. With these simulation parameters, we simply perform binary search for the smallest key ring size that makes the network highly connected. Table 3 shows the simulation results for the EG, q-composite, and the EG with virtual key ring schemes, all with similar parameter settings.

Table 3: Initial key ring size for each sensor node in order to have the deployed WSNs connected with 99.9% probability

Key pre-distribution scheme	Key ring size m	
	Theory	Simulation
EG	176	266
2-composite	279	411
Virtual key ring (EG)	150	158

Observations:

1. The first observation from Table 3 is that our simulation results differ significantly from those predicted by random graph theory. Recall that according to the random graph theory, a network of n nodes is connected with some high probability c if the average node degree $n_t \geq \frac{n-1}{n} \cdot (\ln(n) - \ln(-\ln(c)))$. Theoretically, a network of 1,000 nodes should be connected 99.9% of the time if the expected node degree exceeds 13.8. For a node density of 30, the key ring size required in the EG scheme should be 176, and 279 for the q-composite scheme with $q = 2$. These values are found using the equations in Section 3. However, simulation experiments with these key ring sizes show that the network connectivity is only 95.12% for the EG scheme, and 94.77% for the q-composite scheme. Much larger key ring sizes (about 50% larger) are required to achieve 99.9% connectivity for these two schemes.

These discrepancies are due to (i) the secure connectivity graph in the EG and q-composite schemes is not precisely an Erdős-Rényi random graph, which was observed by Di Pietro *et al.* [14] (they also provided a method to compute the key ring size, however they assume *full visibility* between nodes, which is not applicable in our experiments), (ii) the small number of sensor nodes (1,000) in our experiments, compared to the asymptotic graph theory result, and (iii) the existence of *boundary effects* in the planar region, which limit the connectivity for sensor nodes on the edges and in the corners of the region. Such sensor nodes are at a disadvantage in discovering trust neighbors, and are more likely to be disconnected from the network.

2. *The virtual key ring approach can achieve the desired connectivity with fewer initial keys.* For example, the initial key ring size required in the EG with virtual key ring scheme is about 60% of that required in the original EG scheme, and about 40% of that required in the q-composite scheme.

Since each sensor has a smaller initial key ring, an adversary obtains less information when capturing sensor nodes. That is, the EG with virtual key ring scheme provides greater resilience against node capture. We quantify this result in Section 6.3.

6.3 Resilience against node capture

In this section, we compare resilience of the new virtual key ring scheme with EG in three cases. We first compare the resilience of the basic schemes, and the resilience of the schemes enhanced with multi-path and q-composite techniques, respectively. The simulation results below show the

power of the virtual key ring approach in providing resilience against large-scale node capture.

Virtual key ring scheme: Figure 3 compares the resilience against node capture of the virtual key ring scheme with the original EG scheme. We only consider the strongest adversary A_E who is present also during key establishment. The adversary has the compromised keys and can intercept all the network communications including messages that are sent for key establishment.

Figure 3a shows that the basic virtual key ring technique improves the resilience of EG scheme against A_E by about 10% in large-scale attacks where the number of captured nodes is from 150 to 450. This improvement is mainly because of the smaller initial key rings. The real power of the virtual key ring concept, however, is obtained when it is combined with the multi-path key reinforcement and q-composite techniques.

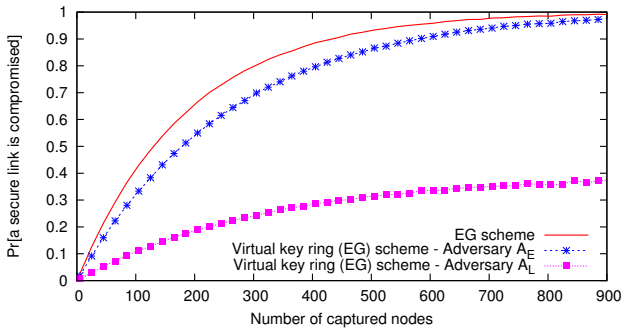
Virtual key ring with multi-path key reinforcement: For efficiency reasons, we assume a restricted version of the multi-path key reinforcement technique in which only paths of length two between nodes are considered for reinforcement. In the virtual key ring scheme with multi-path key reinforcement, after the completion of key establishment, paths of length two are found and portions of a key are sent along each path. A link is considered secure if a link key can be established using the virtual key rings. Since these key rings are much larger, there are many more secure paths compared to the original EG scheme.

Table 4 shows the average node degree in the virtual key ring scheme and the original EG system, for the same connectivity. In the virtual key ring scheme, nodes have higher degree, and so it is expected that the number of common trust neighbors, and hence the number of two-hop secure paths between two nodes, is larger. As a consequence, the probability of at least one path surviving a node capture attack increases, and resilience is improved.

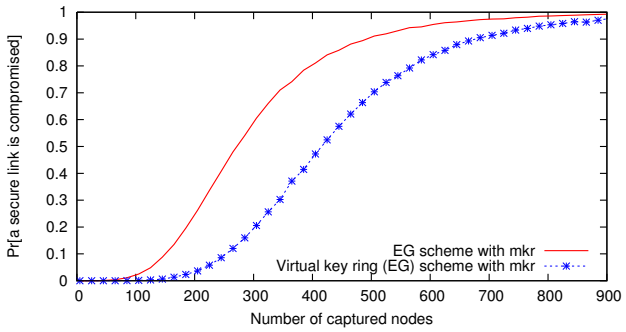
Table 4: The average node degree in the highly-connected WSNs (i.e., connectivity is 99.9%) where the network size is 1,000, the density is 30 and the pool size is 50,000

Key pre-distribution scheme	m	Node degree n_t	
		Theory	Simulation
EG	266	23.6	23.6
2-composite	411	23.6	23.5
Virtual key ring (EG)	158	29.9	29.6

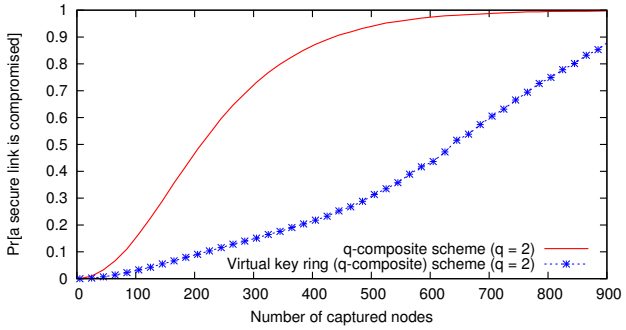
Figure 3b compares simulation results of the resilience of the virtual key ring scheme with multi-path key reinforcement with EG with the same reinforcement. It can be seen that the virtual key ring scheme has much better resilience. For medium size attack when up to 200 nodes (out of 1000) are captured, the virtual key ring scheme is effectively secure – that is, the link compromise probability between uncaptured nodes is less than 3%. This figure for EG scheme grows rapidly for capture of more than 100 nodes and reaches nearly 30% for 200 nodes captured. For 300 nodes captured, the survival probabilities for the two schemes are 80% (virtual key ring) and 40% (EG). For larger captures of up to 500 nodes, the virtual key ring scheme is at least 25% more resilient, and for very large captures of 700 or more nodes, this advantage is around 5-20%. Note that in all cases the network is still functional but the communications are all compromised.



(a) EG vs. Virtual key ring (EG)



(b) EG with multi-path key reinforcement vs. Virtual key ring (EG) with multi-path key reinforcement



(c) 2-composite vs. Virtual key ring (2-composite)

Figure 3: Network size is 1,000, the average number of wireless neighbors is 30, pool size is 50,000 and the key ring size is chosen such that the deployed network is connected 99.9% of the time (based on the simulation results) depending on the scheme.

Virtual key ring with q-composite enhancement: The q-composite enhancement of virtual key ring scheme works as follows. In the initialization phase each node receives an initial key ring. In the first-round key establishment, two wireless neighbors that share at least q ($q \geq 2$) common keys can establish a secure link, and these two nodes are first-round trust neighbors of each other. The link key is the hash value of the concatenation of all the common keys. Each node also constructs a virtual key ring that

consists of its initial key ring and the key IDs of all primary keys of its first-round trust neighbors. In the second round, nodes discover more trust neighbors using their virtual key rings. Two wireless neighbors that have not yet established a secure link and share at least q common keys in their virtual key rings can derive a pairwise key by i) finding a uniquely-generated key share for each common key (i.e., two nodes perform the protocol described in Table 2 repeatedly), and then ii) calculating the hash value of the concatenation of all key shares.

The virtual key rings grow much larger compared to the primary key rings and thus nodes are expected to have many common keys in their virtual key rings. This will make it very difficult for the adversary to construct a second-round link key because of the large number of primary keys that are used in the key generation.

Figure 3c shows the simulation results for the resilience of the virtual key ring enhanced with the q-composite scheme ($q = 2$) and the enhanced version of the original EG. It can be clearly seen that the improved resilience in this case extends to very large captures. The simulation results suggest that even when half of the nodes are captured, a link that is not connected to a captured node has a survival chance of around 70%. This is an impressive result compared to the less than 5% survival chance for the same link in the enhanced EG. For 600–800 nodes captured, the original EG is effectively insecure (less than 2% survival chance) while the virtual key ring scheme still provides a good (30–45%) level of survivability for links.

Discussion: The above simulation results show that the virtual key ring scheme enhanced with multi-path key reinforcement or q-composite techniques can provide survivability against very large-scale node capture attacks. The cost of this survivability is the extra messages that need to be sent regarding keys whose identifiers, and not their actual values, are in the virtual key rings.

7. CONCLUSION

In this paper, we propose a virtual key ring technique to improve resilience and connectivity of random key predistribution schemes. This technique is described and mathematically analyzed as an extension of the EG scheme for demonstration. The results indicate that when applied to the EG scheme, the virtual key ring technique strengthens the resilience against node capture, improves network connectivity, and reduces the memory storage. The cost of establishment of each pairwise key is at most four short messages, and the extra cost of the computation is negligible. We also show how the virtual key ring idea can be applied with the multi-path key reinforcement technique and the q-composite scheme to strengthen the resilience against large-scale node capture attack significantly.

Furthermore, the virtual key ring idea can be combined with threshold-based approaches such as Du *et al.*'s scheme, and Liu *et al.*'s scheme to increase resilience of those schemes against node capture. Our future work includes combining the virtual key ring technique with other key establishment schemes to improve their resilience, examining the effect of more than two rounds of key establishment, and also studying the technique in other network topologies.

Acknowledgments

Financial support for this research was provided by iCORE (Informatics Circle of Research Excellence) in the Province of Alberta, as well as NSERC (Natural Sciences and Engineering Research Council) in Canada.

8. REFERENCES

[1] S. Camtepe and B. Yener, "Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks", *IEEE/ACM Transactions on Networking*, pp. 346-358, April 2007.

[2] D. Stinson, *Combinatorial Designs: Construction and Analysis*, Springer-Verlag, New York, 2004.

[3] D. Carman, P. Kruus, and B. Matt, "Constraints and Approaches for Distributed Sensor Network Security", NAI Labs Technical Report #00-010, September 2000.

[4] H. Chan and A. Perrig, "PIKE: Peer Intermediaries for Key Establishment in Sensor Networks", *Proceedings of IEEE INFOCOM*, pp. 524-535, 2005.

[5] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks", *Proceedings of IEEE Security and Privacy Symposium*, pp. 197-213, May 2003.

[6] W. Du, J. Deng, Y. Han, S. Chen, and P. Varshney, "A Key Management Scheme for Wireless Sensor Networks using Deployment Knowledge", *Proceedings of IEEE INFOCOM*, pp. 586-597, March 2004.

[7] W. Du, J. Deng, Y. Han, P. Varshney, J. Katz, and A. Khalili, "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks", *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 42-51, October 2003.

[8] L. Eschenauer and V. Gligor, "A Key Management Scheme for Distributed Sensor Networks", *Proceedings of the 9th ACM Conference on Computer and Communication Security*, pp. 41-47, November 2002.

[9] D. Huang, M. Mehta, D. Medhi, and L. Harn, "Location-aware Key Management Scheme for Wireless Sensor Networks", *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 29-42, October 2004.

[10] Internet Engineering Task Force, "Diffie-Hellman Key Agreement Method", http://tools.ietf.org/html/rfc2631, 1999.

[11] D. Liu and P. Ning, "Location-based Pairwise Key Establishments for Static Sensor Networks", *Proceedings of ACM Workshop on Security in Ad Hoc and Sensor Networks*, pp. 72-82, 2003.

[12] D. Liu, P. Ning, and W. Du, "Group-based Key Pre-distribution in Wireless Sensor Networks", *Proceedings of ACM Workshop on Wireless Security*, pp. 11-20, 2005.

[13] D. Liu, P. Ning, and R. Liu, "Establishing Pairwise Keys in Distributed Sensor Networks", *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pp. 52-61, October 2003.

[14] R. Pietro, L. Mancini, A. Mei, A. Panconesi, and J. Radhakrishnan, "Redoubtable Sensor Networks", *ACM Transactions on Information and System Security*, 11(3):1–22, 2008.

[15] J. Spencer, "The Strange Logic of Random Graphs",

Algorithms and Combinatorics, Vol. 22, Springer-Verlag, ISBN 3-540-41654-4, 2000.

[16] T. Vu, "Modeling, Analysis, and Simulation of Secure Connectivity in Wireless Sensor Networks", MSc thesis, Department of Computer Science, University of Calgary, October 2009.

[17] T. Vu, C. Williamson, and R. Safavi-Naini, "Simulation Modeling of Secure Wireless Sensor Networks", *Proceedings of ValueTools '09*, Pisa, Italy, October 2009.

APPENDIX

A. SECURITY OF PROTOCOL Π

The security properties of a uniquely-generated key k^* follow from the properties of $E(.,.)$ and $F(.,.)$ as described in Section 4.4.

P1: A passive adversary that is present from the start of protocol can see key establishment messages and has the following view:

$$View(Passive) = \{E(k_{A_i A'_i}, F(k_i, r_i)\}$$

The IND-CPA security of the encryption function ensures that the encryption of $F(k_i; r_i)$ is indistinguishable from a random value and so the adversary does not learn anything about the secondary keys, including k^*, by observing the communication.

P2: An adversary $\mathcal{A}_L$ who has captured a randomly selected subset of nodes, has the following view:

$$View(\mathcal{A}_L) = \{k_i\} \cup \{F(k_j; r_j)\}$$

The set $\{k_i\}$ and $\{F(k_j; r_j)\}$ are the sets of all primary and secondary keys, respectively, that are learned by the adversary $\mathcal{A}_L$. Because of the pseudo-randomness of $F((.,.)$, all values $\{F(k_j; r_j)\}$ with $k_j \neq k$ are independent from $F(k, r)$ since they are outputs of different samples of $F(.,.)$. Also the knowledge of $\{F(k; r_\ell), r_\ell \neq r\}$ does not allow the adversary to learn anything about $F(k, r)$ since these are outputs of a pseudo-random function on different inputs. Finally, even knowing k and a subset $\{F(k, r_i), r_i \neq r\}$ does not let the adversary find the output of the function $F(k, .)$ on an unknown input (r).

P3: An adversary $\mathcal{A}_E$ who has captured a random subset of nodes during the key establishment has the following view:

$$View(\mathcal{A}_E) = \{k_i\} \cup \{r_j, KeyID_j, F(k_j; r_j)\}$$

If k is not in $\{k_i\}$ then using an argument similar to P2 shows that the adversary cannot learn anything about k^*.

B. PROOF OF LEMMA 3

Adversary $\mathcal{A}_L$

All the keys generated in the second round of key establishment are randomly chosen in the key space, and assumed to be unique. Let $\bar{k} = F(k, r)$ represent the key of some second-round secure link, where k is the generation key and r is a random bit string. Only the two endpoint nodes of the corresponding link store $\bar{k}$. In order to compromise some second-round secure link, without capturing either endpoint, the adversary must reconstruct $\bar{k}$ using the generation key k and the original random string r.

Since adversary $\mathcal{A}_L$ does not observe the key establishment process, all the second-round links remain secure. A link is compromised by the adversary if and only if it is a first-round link, and the protection key of this link belongs to some of the captured sensor nodes. The probability that a secure link is a first-round link is $\frac{n_{t_1}}{n_{t_1}+n_{t_2}}$, where n_{t_1} and n_{t_2} are the expected number of first-round and second-round trust neighbors of a sensor node, respectively. Let the protection key of a first-round link be k. The probability that k does not belong to a given initial key ring is $1 - \frac{m}{N}$. Thus, the probability that k does not belong to the x initial key rings from the captured nodes is $(1 - \frac{m}{N})^x$. The probability that k does belong to at least one of the x initial key rings captured is $1 - (1 - \frac{m}{N})^x$.

Hence, by capturing x sensor nodes, a late adversary can compromise a secure link with probability:

$$p_{c_L} = \frac{n_{t_1}}{n_{t_1} + n_{t_2}} \cdot \left(1 - (1 - \frac{m}{N})^x\right)$$

□

Adversary $\mathcal{A}_E$

Adversary $\mathcal{A}_E$ is more powerful than $\mathcal{A}_L$, since $\mathcal{A}_E$ can observe the key establishment process. Assume that the adversary can take full advantage of this, capturing all the random bit strings for generating keys in the second round of key establishment. Thus, a secure link is compromised by $\mathcal{A}_E$ if:

- case e_1: it is a first-round link, and the protection key belongs to some captured sensor node; or
- case e_2: it is a second-round link, and the protection key is generated by some captured sensor node; or
- case e_3: it is a second-round link, and the protection key is not generated by any captured sensor, but the generation key belongs to some captured sensor node.

The probability that an arbitrarily chosen secure link is a first-round link is $\frac{n_{t_1}}{n_{t_1}+n_{t_2}}$ (similarly, $\frac{n_{t_2}}{n_{t_1}+n_{t_2}}$ for the second round). The probability that a random key k in the key pool belongs to one or more of the x initial key rings captured is $1 - (1 - \frac{m}{N})^x$. The challenge is to estimate the probability that the key associated with an arbitrary second-round link is generated by some captured sensor node.

Let A and B be two sensor nodes that have a secure link established in the second round. The probability that the generation key, a common key in their virtual key rings, is not in A's initial key ring is approximately $\frac{m_v - m}{m_v}$. If A does not possess this key, then A has to rely on some first-round trust neighbor A' to generate the unique key for the link AB. The probability that this first-round trust neighbor is captured is $\frac{x}{n}$, where x is the number of captured nodes, and n is the total number of sensors in the network. Therefore, the probability that A obtains the key for the link AB from some captured sensor node is $\frac{m_v - m}{m_v} \cdot \frac{x}{n}$. By symmetry, this is also the probability that B acquires the key from some captured sensor. Assuming independence, the probability that the key of an arbitrary second-round link is generated by some captured sensor node is:

$$1 - (1 - \frac{m_v - m}{m_v} \cdot \frac{x}{n})^2 \quad (8)$$

Given a secure link, we have $\texttt{Pr}[e_1] = p_{c_L}$,

$$\texttt{Pr}[e_2] = \frac{n_{t_2}}{n_{t_1}+n_{t_2}} \cdot \left(1-(1-\frac{m_v-m}{m_v}\cdot\frac{x}{n})^2\right), \qquad (9)$$

and

$$\texttt{Pr}[e_3] = \frac{n_{t_2}}{n_{t_1}+n_{t_2}} \cdot (1-\frac{m_v-m}{m_v}\cdot\frac{x}{n})^2 \cdot \left(1-(1-\frac{m}{N})^x\right) \qquad (10)$$

Thus, if x sensor nodes are captured, the expected fraction of secure links that are compromised by $\mathcal{A}_E$ is: $p_{c_E} = \texttt{Pr}[e_1] + \texttt{Pr}[e_2] + \texttt{Pr}[e_3]$.

□

C. FINDING INITIAL KEY RING SIZE

The initial key ring size m depends on the network size (n), density (n_w), desired connectivity (c), key pool size (N), and the network deployment. In the approximation, the connectivity c is considered to be 99.9%. We perform a binary search for different parameter sets (n, n_w, N) to find the smallest initial key ring size such that the network is connected at least 9,990 times out of 10,000 simulation runs. The simulation parameters are chosen according to the assumptions stated in Section 3. In particular, the network consists of 1,000 nodes or more, and the average node density is 30 to 60. We investigate the pool size in a wide range from 1,000 to 100,000. Furthermore, the deployment region in the simulation is a planar square. The simulation results suggest that the network size has little impact on the initial key ring size. More specifically, when the network size grows from 1,000 to 5,000 nodes and other simulation parameters remain the same, the initial key ring size varies little. Thus, we assume that the initial key ring size is independent of the network size when the network consists of 1,000 nodes or more. In the following discussion, all simulation runs assume a network size of 1,000.

Our goal is to estimate the size of the initial key ring as a function of key pool size and network density. Figure 4 plots the initial key ring size with respect to the key pool size for a fixed network density n_w. The results suggest a linear relationship between $\texttt{log}_{10}(m)$ and $\texttt{log}_{10}(N)$. Assume that this linear relationship is of the form $\texttt{log}_{10}(m) = a \cdot \texttt{log}_{10}(N) + b$. The log-log plot shows the lines all have the same slope regardless of n_w and so a is a constant independent of n_w. We assume b is determined by n_w only.

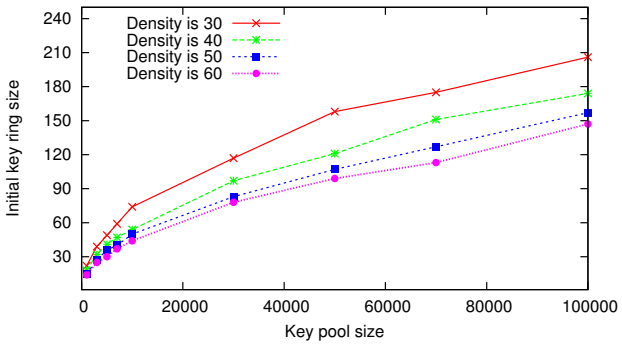
Using *linear least squares* method to find the coefficients for each line and then calculating the average value, we can estimate $a = 0.5$ and $b = 0.0705 - 0.006 \cdot n_w$. The detailed results are available in [16]. This leads to the estimation equation of the initial key ring size as:

$$\hat{m} = 10^{a\cdot\texttt{log}_{10}(N)+b} = 10^{0.5\cdot\texttt{log}_{10}(N)-0.006\cdot n_w} = \frac{\sqrt{N}}{10^{0.006\cdot n_w}}$$

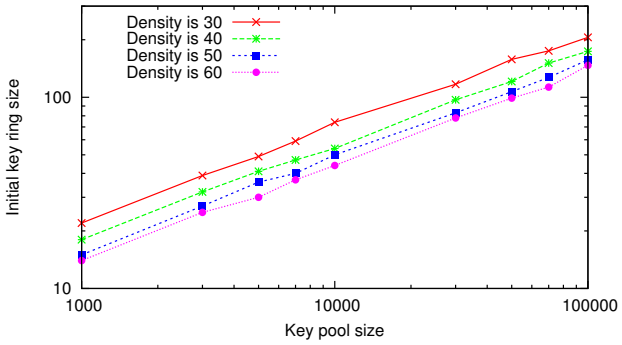
$\frac{1}{10^{0.006\cdot n_w}}$ is further rounded to $\left(\frac{2}{3}\right)^{\frac{n_w}{30}}$ and the final estimation is:

$$\hat{m} = \left(\frac{2}{3}\right)^{\frac{n_w}{30}} \cdot \sqrt{N}$$

□



(a) Linear scale



(b) Log-log scale

Figure 4: Initial key ring size with respect to key pool size plotted in linear and log-log scales. Network size is 1,000.