Optimally Controlled Nonintrusive Broadcasting for Path Key Establishment Mechanism in Wireless Sensor Networks

Aishwarya Mishra
School of Information Technology
Illinois State University
Normal IL 61790 USA
amishra@ilstu.edu

Tibor Gyires
School of Information Technology
Illinois State University
Normal IL 61790 USA
tbgyires@ilstu.edu

Yongning Tang
School of Information Technology
Illinois State University
Normal IL 61790 USA
ytang@ilstu.edu

***Abstract*—Random Key Predistribution Scheme (RKPS) guarantees any pair of neighboring nodes in a Wireless Sensor Network (WSN) can build a secure connection either directly if a common key found, or indirectly through a Path Key Establishment Mechanism (PKEM). When a sensor node resorts to PKEM to establish a secure connection with a neighboring node, it needs to broadcast a keyrequest to all securely connected nodes. However, unbounded broadcasting in PKEM can potentially cause unnecessary or duplicated broadcast message forwarding, which can intrusively incur disruptive power consumption on all involved sensor nodes in a highly resource constrained WSN. Such negative impact can be much worse if exploited by a malicious adversary to launch power exhaustion Denial of Service (DoS) attacks to sabotage a secured WSN. Thus, it is essential to convert unbounded broadcasting in PKEM to a nonintrusive broadcasting with optimally minimal message forwarding boundary in a WSN. Previous research empirically identified bounds on PKEM for small networks, which may not be suitable for densely deployed WSNs with much higher sensor node populations. In this paper, we tackle this problem by applying theoretical results to identifying the upper bound of diameter on a WSN when represented as a Erdős-Rényi random graph. We then verify the performance of a broadcast bounded PKEM through simulations. The performance evaluation shows the effectiveness of the optimally bounded PKEM.**

Keywords- sensor networks; random key predistribution; graph diameter; random graph; theoretical bound.

I. Introduction

A Wireless Sensor Network (WSN) comprises of a large population of inexpensive sensor nodes that form an ad-hoc wireless network to transmit information. The sensor nodes can be deployed in a large geographical area to detect or measure physical quantities such as temperatures, magnetic anomalies, chemicals or motion in their immediate environment. Several important WSN applications require to operate in a hostile environment, where adversaries may attempt to sabotage the WSN via different means, such as unhindered physical access, eavesdropping, message deception triggered exhaustive power consumption.

Securing WSNs is a highly demanded but also highly challenging task, especially when implemented in highly resource constrained sensor nodes. Conventional security mechanism based on public-key cryptography requires extensive computations that are infeasible on current sensor node platforms. Consequently, Symmetric key cryptography has been explored for securing WSNs due to its low computational requirement.

Random Key Predistribution Scheme (RKPS) [2] has been proposed and effectively used to secure WSNs. RKPS relies on predistributing a random subset of keys (keyring) from a large set of keys (keypool) on each sensor node before a WSN is deployed. RKPS uses a small keyring to achieve secure communication between a sensor node and all its neighboring nodes within its transmission range. However, the small keyring size can only be controlled to guarantee that a sensor node is able to authenticate and thus trust a small fraction of its neighboring nodes after its deployment. If the sensor node is unable to find a common key with a neighboring node, it will broadcast an authentication request (keyrequest) to all its trusted nodes who could either authenticate the incoming keyrequest or forward the request to their trusted sensor nodes for authentication. The process will continue until the keyrequest is authenticated.

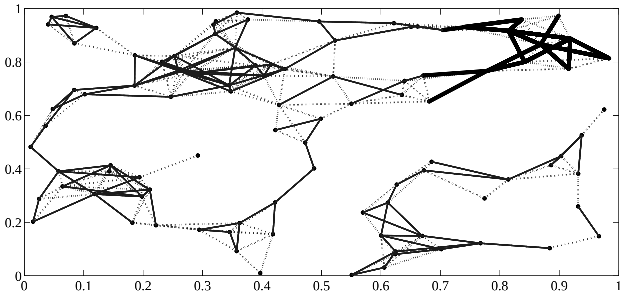

Figure 1: PKEM in progress over a RKPS secured WSN.

The keyrequest forwarding in RKPS is analogical to a flooding broadcast mechanism. Optimally bounding keyrequest broadcasting in Path Key Establishment Mechanism (PKEM) [2] is the focus of this paper. We discuss this mechanism further in Section III while detailing a general model of basic RKPS.

Figure 1 shows a RKPS enabled WSN, where (1) thick links (as shown on the upper right corner) represent indirect secure connections created by PKEM, (2) the solid lines represent direct secure connection between two neighboring sensor nodes with at least one common key, and (3) the dashed lines represent unsecured connections between two sensor nodes without a common key. Sensor nodes connected by dashed lines can resort to PKEM to authenticate each other and construct secure connections indirectly. The diagram plots the Cartesian coordinates of sensor node locations that uniformly distributed on a unit square to represent a sensor field.

RKPS can be used to secure a WSN using a limited keyring size with the cost of network communication overhead, which grows significantly in large-scale WSNs. Thus, it is desirable to control the communication overhead in PKEM, especially if the sensor nodes are allowed to forward a keyrequest unconditionally. For handling a keyrequest, a sensor node needs to consume its limited battery power for receiving, processing (e.g., searching common keys) and forwarding or acknowledging the request. In PKEM, a keyrequest can be initiated by any sensor node, which might be exploited by an adversary to inject bogus keyrequests. In response, a large number of sensor nodes in the attacked WSN will potentially consume a large amount of power to authenticate the bogus keyrequests being injected. Such an exploited weakness in PKEM can easily result to a Denial-of-Service (DoS) attack, which can easily exhaust the battery power on many involved sensor nodes.

The communication overhead in PKEM can be effectively controlled through the use of a Time-To-Live (TTL) parameter on each packet. However, the TTL value needs to satisfy multiple conflicting constraints. Firstly, the TTL needs to be sufficiently large to ensure that (1) PKEM can perform its intended function, and (2) every keyrequest can be forwarded to a sensor node that can authenticate it. On the other hand, this TTL should be as small as possible to limit the communication overhead in PKEMS. Furthermore, a mechanism should exist to enforce that the TTL value does not exceed a value decided before the deployment of a WSN. This is to ensure that an adversary should not be able to inject keyrequest into the WSN with arbitrarily (large) TTLs to launch the DoS attack. Finally, the TTL value should be adaptive for the varying size of a WSN due to newly added nodes or gradual death of existing nodes.

In this paper, we show how to model a secured WSN as a connected Erdős-Rényi random graph such that a keyrequest in RKPS originated from any sensor node can be guaranteed to reach every other sensor node within the WSN. The keyring and keypool sizes are chosen to ensure that a sensor node connects securely to its neighbor nodes with a predictable probability, which can further ensure the resulting graph is connected.

In our previous work [1], we introduced the problem of identifying the maximum TTL (MAXTTL) and applied the related results from Erdős-Rényi random graph theory to identify MAXTTL for a WSN with full-visibility. More specifically, the diameter of the Erdős-Rényi random graphs can be used to calculate the maximum value of the TTL. If one calculates the shortest paths between every pair of nodes within a random graph, the diameter would be the longest of these shortest paths. The full visibility case, discussed further in the next section describes a deployment of the sensor network where every sensor node can connect to any other sensor node within the network. In this paper we expand upon our conference paper to give more details about the problem, our simulation approach and results.

The rest of the paper is organized as the following. Section II discusses the related work. Section III provides the background of PKEM and derives the theoretical bound of flooding radius in PKEM. Section IV presents our simulation design and results. Finally, Section V and Section VI present some discussion and conclude the paper.

II. Related Work

In this section, we will review the work that has been proposed to obtain a suitable TTL

Basic RKPS was first introduced in [2] and its security characteristics have been extensively studied. A variety of schemes have built upon the basic RKPS by combining it with other key predistribution schemes for improving its resilience to node compromise. Reference [4] reviews the basic RKPS and its derivative schemes and also surveys the state-of-art in sensor network security. PKEM overhead applies to the basic RKPS and all its derivative schemes that trade network overhead for reduced keyring size. Since our work is fundamental to RKPS itself and also remains a component of its derivatives, we will work with a model of the Basic RKPS that includes all the elements of the scheme that have remained invariant in its derivative schemes also. Instead of reviewing the specifics of each scheme based on RKPS we dedicate this space to reviewing research that has investigated the characteristics of PKEM or has provided some guidance on its possible values.

The original work in [2] reported empirical observations that the keypath length did not exceed a constant number for the range of node populations between 1000 and 10000, in their simulations. However it did not provide analytical guidance on how PKEM will behave for larger node populations and different node neighborhoods. The first reference to use a TTL limited PKEM appears in [5], where it was recommended to set the keypaths based on the average path lengths in the trust graph. This conclusion was based on empirical observations on an experiment setup similar to the original work in [2]. It was also noted that a majority of the keypath lengths were much smaller than the observed average and the maximum. However, it did not analytically characterize the asymptotic behavior of the PKEM path-lengths and how it evolves with node population, deployment density or average node connectivity. Another contribution of [5] was the explicit statement of the assumptions related to the minimum degree of the underlying connectivity graph. The node connectivity in the original work and many of the derivative schemes has been assumed to be much higher than that supportable by state-of-art MAC layer protocols such as IEEE 802.15.4 Zigbee [11] standard, popular on several sensor node platform implementations.

Apart from empirical observations related to TTL, interesting progress has been made in the investigation of the validity of Erdős-Rényi Random Graphs [7] that discussed the application of graph theory to RKPS in the context of sensor networks and produced validating results for specific ranges of its parameters.

The work in [12] applies random graph theory to RKPS to propose a graph theoretic framework for parametric decision making for RKPS, optimal keyring size, and network transmission energy consumed in PKEM etc. It provides some analytical formulations on the basis of the diameter of de Bruijn's graphs [27] but did not provide any analytical guidance on how to find the diameter of a RKPS trust graph. Our work may be considered supplementary to this research since reliable estimation of the diameter of RKPS trust graph may allow exact derivation of some of the quantities mentioned in this work. Our work instead focuses on proposing supporting theory on the asymptotic bounds of the diameter of the sensor network configuration, which relates it to RKPS parameters such as node population, probability and node neighborhood.

We also note the recent theoretical investigation in Uniform Random Intersection Graphs that model Random Key Predistribution Scheme under the full-visibility assumption. Related work in [23]–[25] investigate several interesting properties of Uniform Random Intersection Graphs and formally prove its connectivity properties and node degree. Finally, we note that [22] and [26] study the diameter of Uniform Random Intersection Graph and solves a problem very similar to ours from a theoretical point of view. We could have chosen to base our analytical model on the basis of [22], however, the focus of this paper is inclined towards the investigation and extension of Erdős-Rényi random graph theory as applied to RKPS implementation on sensor networks.

For the construction of our simulator we used the guidance from [6] that discusses the construction of a high performance simulation for Key Predistribution Schemes for WSN in Java. Our experiment design replicates the experiment set up in [2], and we simulated node populations between 1000 to 10000 sensor

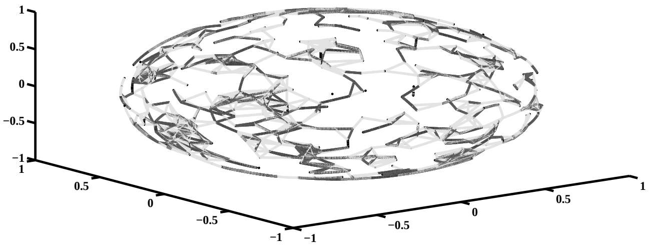

Figure 2: Trust Graph Illustration: Lighter edges represent wireless connectivity and darker edges represent secure connectivity.

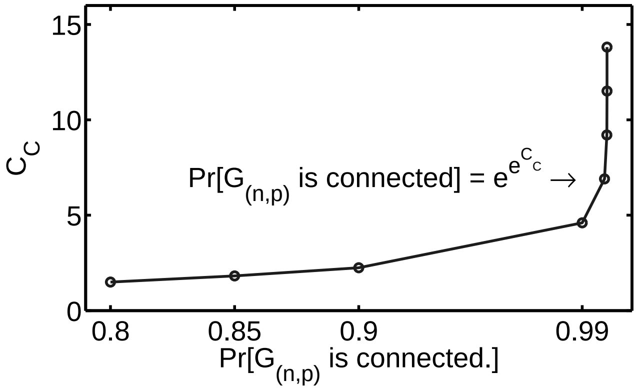

Figure 3: Variation of C_c for desired probability of graph connectivity in Eq. 3.

nodes. Finally, we used a number of open source software libraries for high precision arithmetic, graph theory algorithms and MATLAB for experiment design, statistical analysis and visualization. We discuss these in further detail in Section IV.

III. Theoretical Bound Analysis

For a given WSN with node population n, RKPS models it as a Erdős-Rényi random graph such that appropriate keyring size k and keypool size K can be selected to ensure the formed secure network remain connected.

A. Trust Graph

A sensor network can be modeled as a pair of overlaid graphs, where the underlying graph represents the wireless connectivity among sensor nodes, and the overlay graph shows the secured wireless connectivity among sensor nodes. In the underlying wireless connectivity graph, each sensor node is denoted as a vertex and each wireless connection between two neighboring nodes is represented as a link. In the overlay secured wireless connectivity graph, also referred as Trust Graph, only secured wireless connections remain and are represented as secured links. Figure 2 shows a trust graph overlaid on the top of the connectivity graph of a WSN deployed on the surface of a unit sphere. A trust graph is a sub-graph of the underlying wireless connectivity graph, since its edges only exist if an underlying connectivity graph edge exists. The figure plots the Cartesian coordinates of sensor node locations distributed on the surface of a model sphere, with unit radius. We discuss this modeling approach further in Section IV, Deployment Model.

B. Generalized RKPS Model

In this section, we describe the general model of basic RKPS [2].

Before a WSN is deployed, RKPS allocates a small random subset of keys (keyrings) on each sensor node from a large universal set of random keys (keypool), where each subset may overlap with other subsets with a small probability p. Once deployed, each sensor initiates a shared key discovery protocol with its neighboring nodes by sending a keyrequest containing unique identifiers for each key in its keyring. The neighboring node with common key will respond back by encrypting a random number with a common key (challenge), which will be decrypted by the requesting node and sent back (response) to complete the authentication process. Subsequently, the identified common key can be used to negotiate a shared session encryption key.

The small keyrings only allow a fraction of neighboring nodes to directly authenticate the received keyrequest. A sensor node that unable to authenticate a targeted neighboring node would resort to a path key establishment mechanism (PKEM), where it forwards the keyrequest to its authenticated and thus trusted neighboring nodes. These neighboring nodes would either authenticate the targeted node or forward it to their trusted neighboring nodes until a transitively trusted node authenticates the targeted neighboring node.

The deployment model of a WSN is generally assumed to be uniformly random, and thus the neighboring nodes of any given sensor node cannot be predicted. An Erdős-Rényi random graph is denoted as $G_{(n,p)}$, where n is the number of vertices and p

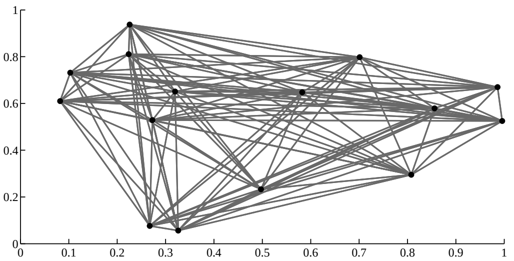

Figure 4: Sensor Network Model with full-visibility assumption.

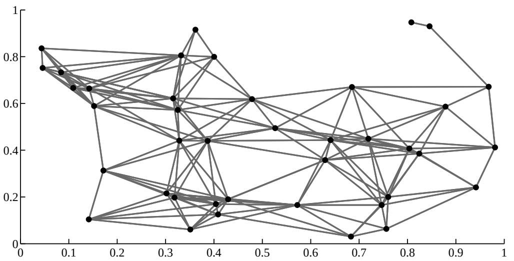

Figure 5: Sensor Network Model with limited-visibility assumption.

represents the probability that a vertex is connected to any other vertex within the graph.

For a random graph $G_{(n,p)}$, we have

$$if \quad p = \frac{\ln(n)}{n} + \frac{C_c}{n} \tag{1}$$

$$then \lim_{n\to\infty} P(G_{(n,p)} \ is \ connected) = e^{e^{-C_c}} \tag{2}$$

where C_c is a constant and should be chosen such that $P(G_{(n,p)} \ is \ connected)$ is close to 1.

Prior research [2] on RKPS has recommended choosing the value of C between 8 and 16, as shown in Figure 3 which can yield the desired value of p, and further derive the keyring size (k) for a given keypool size (K).

It is essential to note that the Erdős-Rényi graph theory assumes that within the graph any node can be connected to another one, i.e., every node can see any others within the network (full-visibility model). However, in sensor networks a sensor node is only connected to a small subset $n_a : n_a \ll n$ of the randomly deployed nodes that are within its transmission range (limited-visibility model). Figure 4 visually illustrates a WSN modeled under the full-visibility assumption in Erdős-Rényi random graph theory. Figure 5 shows a sensor network modeled under the limited-visibility, encountered in practical sensor network deployments. In both, Figure 4 and Figure 5, the nodes represented by the plotted points are only joined by edges if there is connectivity between them. The dimensions represent distance and the plot models the Cartesian coordinates of each sensor's deployment location on a two dimensional square sensor field. Note that in case of full-visibility Figure 4, each node is connected to every other node and its location is immaterial. In contrast, the location and neighborhood of a sensor in the limited-visibility Figure 5 case governs its connectivity with other sensors in the network.

In order to overcome the lack of connectivity of the limited-visibility case, the work in [2] proposed adjusting p to the effective probability (p_a), with which a node can connect to any of its neighboring nodes, such that the average degree d of the nodes in the graph remains constant as shown by Eq. 3.

$$d = (n_a - 1)p_a = np \tag{3}$$

With this calculated value of p_a, the work in [2] derived k according to the following equation:

$$p_a = 1 - \frac{(K-k)!^2}{K!\,(K-2k)!} \tag{4}$$

Research results identifying the upper bound on the random graph diameter with the parameter C_C controlled with the proposed range have been proposed in Theorem 4 in [8], where we have $p \geq c\ln(n)/n$.

C. Diameter of a Sparse Random Graph

Several studies have analytically investigated the upper bound of Erdős-Rényi random graphs for various ranges of n and p. For example, the work presented in [8] reviews the analytical results on various ranges of p, in terms of n. Moreover, it derives the asymptotic bounds on the diameter of Erdős-Rényi random graphs at its critical threshold where both n and p satisfy the relationship mentioned in Eq. 1. Theorem 3 in [8] states that given the relationship in Eq. 5, the diameter of the graph is concentrated on at most three values around value indicated in Eq. 6.

$$\frac{np}{\ln n} = c \geq 2 \tag{5}$$

$$diam(G_{(n,p)}) \leq \lceil \frac{\ln n}{\ln np} \rceil + 1 \tag{6}$$

Furthermore, for ranges of $c \leq 2$, Theorem 4 [8] predicts the upper bound on the diameter indicated by Eq. 7 as follows.

$$\begin{aligned} \left\lceil \frac{\ln(\frac{cn}{11})}{\ln(np)} \right\rceil &\leq diam(G_{(n,p)}) \\ &\leq \left\lceil \frac{\ln(\frac{33c^2}{400})n\ln(n)}{\ln(np)} \right\rceil + 2\left\lfloor \frac{1}{c} \right\rfloor + 2 \end{aligned} \tag{7}$$

The above formula gives the upper-bound on the diameter of sparse random graph, where $p \geq c\ln(n)/n$. It worths noting that c in Eq. 7 can be greater than 2 since it depends upon the value of C_c as shown in Eq. 1. Figure 7 shows how c varies with the value of C_c. A sufficiently high value of C_c can be chosen to get $c \geq 2$. Our experiment design takes this precondition into account to interpret the observed results.

IV. Simulation Design and Results

To investigate the effective diameter of a trust graph for the RKPS configurations discussed in Section III, we created a sensor network simulator along the directions discussed in [11]. Our simulator derives the keying size based on [2] as discussed in Eq. 4, and allows for reasonable variations in the sensor network deployment densities.

A. Simulation Model

The simulation has been constructed in MATLAB and Java. The implementation of the WSN model in Java allows us to take advantage of efficient thread-safe data structures to model a WSN, sensor keyrings and sensor nodes. We have implemented several different experiments using MATLAB to collect the data from Java simulation and interpret it to construct the result visualizations. With the support of several MATLAB toolboxes, we were able to implement efficient graph algorithms for the calculation of all-pair shortest paths and the size of a WSN.

As shown in Figure 6, our simulation model comprises of a MATLAB driver script that calls multiple MATLAB functions to prepare simulation arguments and pass them to the Java simulation model. At the completion of the simulations, the state of the Java simulation is imported into Matlab in the form of an adjacency matrix representation of the trust graph and the connectivity graph.

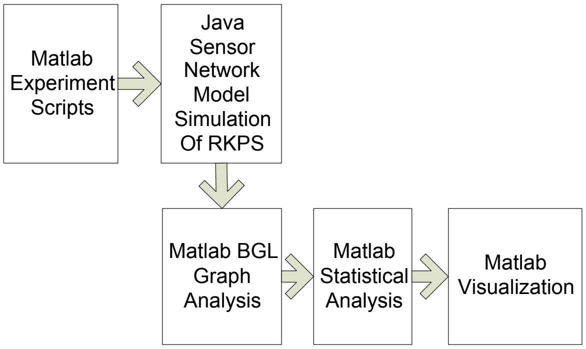
Matlab Experiment Scripts
Java Sensor Network Model Simulation Of RKPS
Matlab BGL Graph Analysis
Matlab Statistical Analysis
Matlab Visualization

Figure 6: Simulation Model.

Adjacency Matrix is a widely used standard representation of a graph in many graph theory algorithms. Informally, the adjacency matrix is a square symmetric matrix with n rows and n columns, where n represents the number of nodes within the graph. Each row corresponds to a node within the graph and contains a value of 1 for each edge that connects that node to another node in the graph. The adjacency matrix can be used as input for the all-pair shortest path graph algorithms available in MATLAB-BGL [20].

MATLAB-BGL provides a MATLAB wrapper for the standard Boost Graphics Library (BGL) [9]. BGL is a comprehensive C++ library implementing almost all known efficient graph algorithms. BGL is well-established and well-reviewed by the development community, and becomes the standard library used for graph theory calculations.

The diameter of a trust graph is the longest path among all shortest paths between any pair of vertices in the graph. Various so-called all-pair shortest path algorithms allow us to measure the shortest number of hops between every pair of sensor nodes in the network. The choice of a suitable algorithm to identify all-pairs shortest paths depends upon the sparseness of the graph and the space complexity of an algorithm with a given graph size.

Assuming V to be the number of vertices and E to be the number of edges in a graph respectively, the complexity of Johnsons [18] all-pair shortest path algorithm is $O(V^2 \ln V + VE)$. Floyd-Warshall [17] provides another approach with the complexity of $O(V^3)$, which is independent of the number of edges. Floyd-Warshall algorithm is generally well-suited for dense graphs with a large number of edges. However, it requires matrix multiplication that is relatively difficult to implement for large-scale networks utilizing built-in data types of common programming languages like C++ and Java. Johnsons algorithm on the other hand has lower space complexity and is therefore well-suited for calculation of the all-pair shortest paths for large-scale networks, even when the number of calculations required for dense graphs may be higher than for Floyd-Warshall. We utilized BGL implementation of Johnsons all-pair shortest paths to calculate the diameter of networks in the range of 1000 to $10,000$ nodes.

We designed experiments with various ranges of n and C_c to validate whether the obtained trust graph from simulations follows the theoretical results. We generated random topologies for creating WSNs by varying the number of nodes from 1000 to 10000. The required probability to obtain full connectivity were calculated based on Eq. 1 and Eq. 3.

B. Verification of Theoretical Results

We utilized the Boost Graph Library [9] and Matlab-BGL [20] toolkit for MATLAB to verify the theoretical results on several instances of random graphs for various values of n when $1 \leq c \leq 2$. Our simulation results as presented below confirm the theoretical results as shown in Section III based on full visibility RKPS models with a keypool size of 100000.

Please note that the diameter values remain relatively stable for large increments of n, which should allow the future extension of a WSN, even with the current controlled diameter. We also observe that the observed value is well-below the value predicted by the theory, which would make it robust against transmission failures in the shortest path.

As discussed earlier, most empirical studies of RKPS have assumed a value of C_c in the range of 8 to 16. Figure 7 and Figure 8 plot the value of c as in Eq. 5 and Eq. 6 showing that c can be assumed to be higher than 2 for lower ranges of n and higher ranges of C_c as in Eq. 1. These values are coincident with the range assumed in prior research on RKPS schemes.

The value of C_c in Eq. 1 has significant impact upon weather c in Eq. 5 is in a range where the diameter of the random graph remains $O(\ln(n)/\ln(np))$. Figure 7 implies that lower values of C_c in Eq. 1 will not allow the diameter of the graph to remain small.

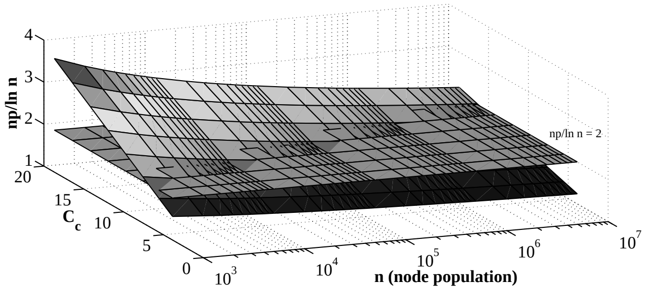

Figure 7: Plot of $np/\ln(n)$ showing the value of c in Eq. 5 for various ranges of n and C_c.

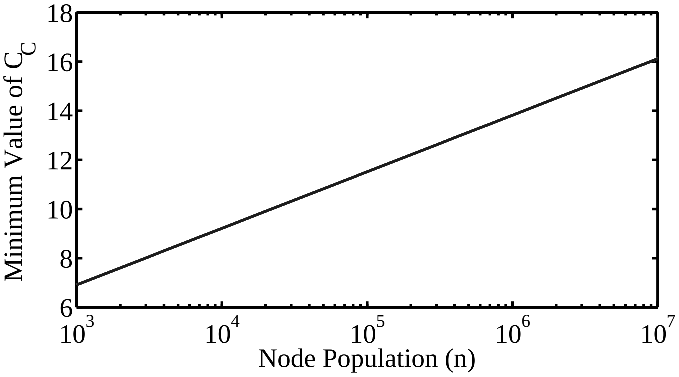

Figure 8: Plot showing $C_n \geq \ln(n)$, values of C_c where $np/\ln(n) = 2$.

C. Keyring Size Calculation

The keyring size was calculated for a keypool of 100000 on the basis of [2] as described by Eq. 5. To work with factorials of large numbers as required by Eq. 5, we utilized the JScience scientific library to support numbers of arbitrary precision. Further, to improve the precision and the performance of our simulations, we created a symbolic fraction that allows canceling of factors in numerator and denominator of a fraction before calculation of its final value. This reduces the loss of precision due to floating point arithmetic operations and enabled us to reproduce the exact calculations for keyring sizes as published in [2]. Since the presence of factorials in Eq. 5 does not permit further simplification for obtaining keyring size through a formula, we utilize a hit and trial approach for calculation of the keyring size for a given keypool size.

To speed up the simulations, we observe that the increase in keyring size would result in higher probability of connectivity. We exploit this monotonicity

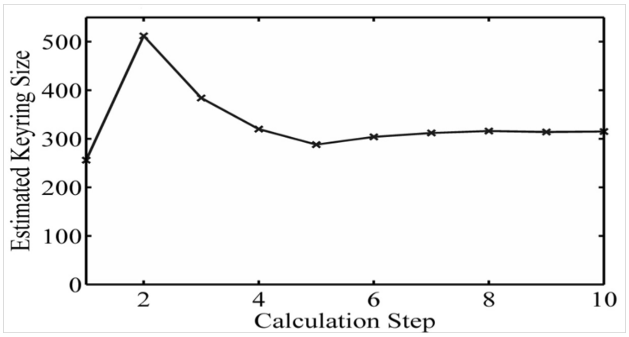

Figure 9: Calculation steps for a keypool size of 10^5 with desired probability of 0.63.

of Eq. 5 to devise a simple binary search to converge on the keyring size with the least number of trials. Figure 9 shows the calculation steps required for typical values of probability and keypool size. Firstly, we rapidly increase keyring sizes in steps of 256 to obtain a probability higher than the required; then we gradually alternate between increasing and decreasing the keyring size to converge upon the size of the keyring, which would give the exact desired probability of connectivity for a given keypool size.

D. Deployment Model

Basic RKPS simulation model uses a unit square as the deployment area. Node density was assumed to be uniform and simulated by varying the transmission range of the sensor node model. More recent work in [11] showed the boundary effect in the context of simulating key predistribution schemes for WSNs. Boundary effect occurs at the borders of the sensor network, where the sensor nodes do not have the average neighborhood connectivity as available to nodes closer to the center. Boundary Effect can significantly influence the degree distribution of the trust graph in simulations but its impact in practical deployments is considerably less as the network grows larger. A recommended elimination [11] of this effect is to modify the WSN deployment model on a spherical surface, which results in a uniformly distributed node population in a WSN. Eliminating the boundary effect also allows us to produce a sensor network model with homogeneous node connectivity, which can be further mitigated if the boundary nodes resort to dynamic range extension as suggested by [5].

To simulate a spherical deployment field, we followed the directions from the work in [7], and modeled our node deployment using Ziggurat method due to Marsaglia [13]. This method allowed us to generate a uniform distribution of three dimentional points on the surface of a sphere. We calculated the node distances using the great circle arc length, with assumption that the node range is a disk shaped area on the surface of the sphere. This is equivalent to the transmission range of a sensor node on a planner surface.

V. Results and Discussion

Figure 10 shows a plot of our simulations on MATLAB, where the diameter of the generated random graph closely follows the the theoretical expectation as described in Section II. The theoretical predictions of the figure is a composite generated on the basis of Eq. 6 and Eq. 7. For the points that satisfy the precondition in Eq. 5 we have used Eq. 6 and Eq. 7 for the points for the others. As shown in Figure 10 the practical diameter of the trust graph is co-incident to the theoretical expectations with an error of ± 1.

Figure 11 and Figure 12 show the long range predictions of the analytical tools that we have discussed in this paper. The predict shows that the diameter of a WSN will increase very slowly with the increase of network size, and will remain constant for large ranges of node populations. This further shows that setting a maximum limit to the TTL employed by PKEM will not interfere with the extensibility of the sensor network. More sensor nodes can be deployed later with the same TTL setting to continue operation of the network. Moreover, the stability of TTL for largely varying network sizes also show that the network will be robust against failures or compromise of a large percentage of sensor nodes, and PKEM operation will not be impacted by a limited TTL. On the other hand, this also indicates that controlling TTL would only provide limited control over the number of nodes visited by a keyrequest and the consequent power consumption of PKEM. The number of nodes that may receive a PKEM request rises rapidly with the increment of TTL in a large-scale WSN.

Figure 12 shows the node degrees may rise as high as 140, which is prohibitively high for current sensor node platforms. We notice that several methods have been proposed to mitigate this problem, including range extension [5]. Further investigation of the diameters of practical sensor network deployments should be undertaken using simulations and analytical models,

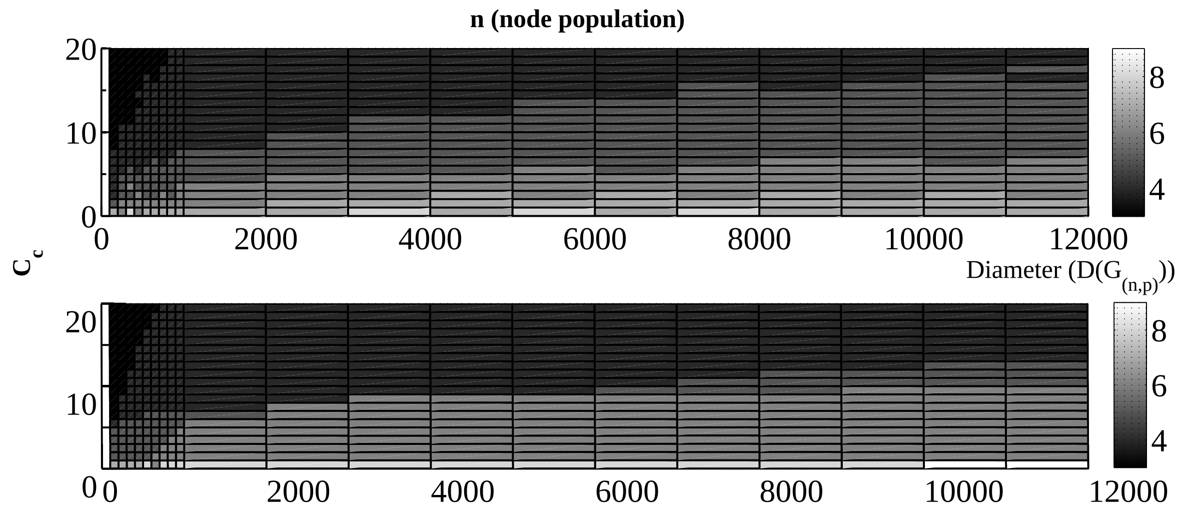

Figure 10: The comparison of practical and theoretical graph diameter.

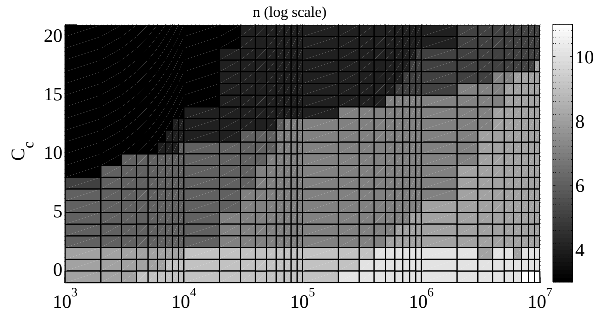

Figure 11: Log scale plot of diameter for large network sizes n and C_c.

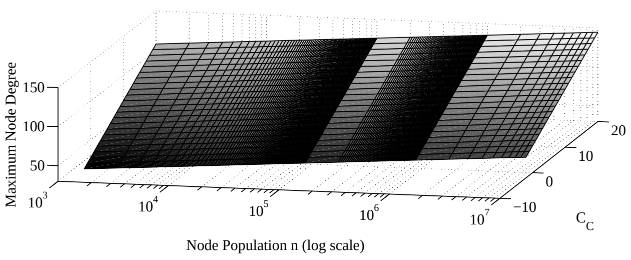

Figure 12: Log scale plot of maximum node degree for large network sizes n as predicted by [10].

which specifically address the limited-visibility sensor network deployment model, such as Random Geometric Graphs(RGG) [16].

VI. Conclusion and Further Work

To conclude, this paper formally studies the communication overhead of PKEM and its possible improvement through Erdős-Rényi random graph theory. PKEM employs a variant of flooding broadcasting and specifically an instance of probabilistic broadcasting [19]. We have shown that the theory on the diameter of the Erdős-Rényi random graph can be used to limit the overhead of the PKEM without impacting its function in RKPS. While we have focused on PKEM specifically, the key revocation protocol for RKPS also relies on broadcasting. Thus, our results can be directly applied to limit the overhead of the key revocation protocol also.

We have presented and tested an analytical model that provides a simplified guidance on the TTL setting in PKEM for sensor network deployments under full visibility setting. We have shown that certain assumptions regarding the modeling of the trust graph are necessary to preserve its properties as applicable in an Erdős-Rényi random graph. Lastly, we have studied the predictions of our analytical model for large scale deployment and identified their impact on the feasibility of large scale sensor networks.

In this paper we have studied the solution of the MAXTTL problem for the full-visibility case where a sensor can potentially communicate (see) with any other sensor within the network. A majority of practical sensor network deployment confirm to the limited visibility case where a sensor can only communicate (see) other sensor nodes within its transmission range. We intend to extend this work for further for practical sensor network deployments under the limited visibility assumption. Recent work in modeling practical sensor networks deployments have utilized Random Geometric Graph theory. We intend to explore the theoretical results on Random Geometric Graphs to find guidance on the diameter of a RKPS trust graph under the limited visibility assumption.

We chose this problem to trigger a discussion of the energy consumption of RKPS when PKEM transmissions are also taken into account. Optimally controlling the transmission overhead in RKPS is critical to its eventual success as a security scheme for WSNs. Competing public-key cryptography schemes generally require much smaller number of transmissions, and may eventually become viable on somewhat more powerful sensor node platforms. Finally, bounded keyrequest broadcasting and methods to securely limit its overhead in RKPS are essential to mitigate adversarial DoS attacks. These DoS attacks are not defendable because RKPS cannot identify weather a keyrequest originates from an authentic sensor node or an adversary. For achieving this function, it would require an authentication scheme that is at least as secure as RKPS, preferably with lesser overhead.

Randomized broadcasting (or gossiping) has been considered as another method to lower the transmission complexity of RKPS, and may be more suitable for implementation on some sensor network configurations and node populations. However, PKEM based on randomized broadcasting trades latency and reliability for lower transmission complexity. Finally, we expect to provide a skeleton of theoretical assumptions, which may facilitate the application of results in Erdős-Rényi random graph theory to the problem of broadcasting at large, and the application adopting its upper bound on diameter in bounding the TTL values for flood broadcasting at large.

Our work also shows that the secure connectivity and diameter of the trust graph is intimately related to the deployment density of a WSN, and the average node connectivity. A poorly connected graph would result in a sparser trust graph, and may result in unreliable operation of PKEM with a limited TTL. Sparser trust graphs may require PKEM to broadcast packets with higher TTL values that exposes it to undesired potential DoS attacks. A predefined upper bound and sensor network configuration with a smaller diameter would effectively prevent an adversary from exploiting this mechanism.

References

[1] A. Mishra, T. Gyires and Y. Tang, "Towards A Theoretically Bounded Path Key Establishment Mechanism in Wireless Sensor Networks," in Proceedings of the IARIA ICN 2012, The Eleventh International Conference on Networks, Saint Gilles, Reunion, 2012.

[2] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA, 2002.

[3] P. Erdős and A. Rényi, "On the evolution of random graphs," Magyar Tud. Akad. Mat. Kutat Int. Kzl, vol. 5, pp. 17-61, 1960.

[4] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway, "A survey of key management schemes in wireless sensor networks," Comput. Commun., vol. 30, 2007, pp. 2314-2341.

[5] J. Hwang and Y. Kim, "Revisiting random key predistribution schemes for wireless sensor networks," in Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks, Washington DC, USA, 2004, pp. 43-52.

[6] T. M. Vu, R. Safavi-Naini, and C. Williamson, "On applicability of random graphs for modeling random key predistribution for wireless sensor networks," in Proceedings of the 12th International Conference on Stabilization, Safety, and Security of Distributed Systems, NewYork, NY, USA, 2010.

[7] V. Tuan Manh, W. Carey, and S.-N. Reihaneh, "Simulation modeling of secure wireless sensor networks," in Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools, Pisa, Italy, 2009, pp. 1-10.

[8] F. Chung and L. Lu, "The Diameter of Sparse Random Graphs," Advances in Applied Mathematics, vol. 26, pp. 257-279, 2001.

[9] J. G. Siek, Lie-Quan Lee, and Andrew Lumsdaine. "The Boost Graph Library: User Guide and Reference Manual": C++ In-Depth Series. Addison-Wesley Professional, December 2001.

[10] R. Durrett, Random Graph Dynamics. New York, NY: Cambridge University Press 2006.

[11] J. Zheng and M. J. Lee, "A comprehensive performance study of IEEE 802.15. 4," Sensor network operations, 2004, pp. 218-237.

[12] A. C. F. Chan, "A graph theoretic approach for optimizing key pre-distribution in wireless sensor networks," in Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2009. WiOPT 2009. 7th International Symposium on, 2009, pp. 1-8.

[13] G. Marsaglia and W. Tsang, The Ziggurat method for generating random variables, 2000.

[14] B. Bollobás, Random Graphs: Academic Press, London, 1985.

[15] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in Advances in cryptologyCRYPTO92, 1993, pp. 471-486.

[16] M. Penrose, Random geometric graphs: Oxford University Press, 2003.

[17] M. J. Atallah and M. Blanton, Foundations of Algorithms and Theory of Computation: Taylor & Francis, 2009.

[18] Paul E. Black, "Johnson's Algorithm", Dictionary of Algorithms and Data Structures, National Institute of Standards and Technology 2004[12/01/2012].

[19] Y. Sasson, et al., "Probabilistic broadcast for flooding in wireless mobile ad hoc networks," 2003, pp. 1124-1130 vol. 2.

[20] D. Gleich, "MatlabBGL. A Matlab Graph Library," Institute for Computational and Mathematical Engineering, Stanford University. Available: http://www. stanford. edu/ dgleich/programs/matlab˙bgl, 2008[12/01/2012].

[21] P. Roberto Di, V. M. Luigi, M. Alessandro, P. Alessandro, and R. Jaikumar, "Redoubtable Sensor Networks," ACM Trans. Inf. Syst. Secur., vol. 11, pp. 1-22, 2008.

[22] K. Rybarczyk, "Diameter, connectivity, and phase transition of the uniform random intersection graph," Discrete Mathematics, vol. 311, pp. 1998-2019, 2011.

[23] M. Karosńki, et al., "On Random Intersection Graphs: The Subgraph Problem," Comb. Probab. Comput., vol. 8, pp. 131-159, 1999.

[24] S. R. Blackburn and S. Gerke, "Connectivity of the uniform random intersection graph," Discrete Mathematics, vol. 309, pp. 5130-5140, 2009.

[25] M. Deijfen and W. Kets, "Random intersection graphs with tunable degree distribution and clustering," Probab. Eng. Inf. Sci., vol. 23, pp. 661-674, 2009.

[26] O. Yağan and A. M. Makowski, "Random Key Graphs Can They be Small Worlds?," in Networks and Communications, 2009. NETCOM'09. First International Conference on, 2009, pp. 313-318.

[27] N. G. de Bruijn and P. P. Erdős, "A combinatorial problem," Koninklijke Netherlands: Academe Van Wetenschappen, vol. 49, 1946, pp. 758-764.