Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 17

Andreas U. Schmidt Shiguo Lian (Eds.)

Security and Privacy in Mobile Information and Communication Systems

First International ICST Conference, MobiSec 2009
Turin, Italy, June 3-5, 2009
Revised Selected Papers

Springer

Volume Editors

Andreas U. Schmidt
novalyst IT AG
Robert-Bosch Str. 38
61184 Karben, Germany
E-mail: andreas.schmidt@novalyst.de

Shiguo Lian
France Telecom R&D
10F, South Twr., Raycom Infotech, Park C, 2
Science Institute South Rd.
Haidian District, Beijing 100080, China
E-mail: shiguo.lian@orange-ftgroup.com

Library of Congress Control Number: 2009935010

CR Subject Classification (1998): C.2, D.4.6, K.6.5, K.4.4, C.1.3, K.4.1

ISSN 1867-8211
ISBN-10 3-642-04433-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-04433-5 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12761173 06/3180 5 4 3 2 1 0

Preface

MobiSec 2009 was the first ICST conference on security and privacy in mobile information and communication systems. Within the vast area of mobile technology research and application, the intention behind the creation of MobiSec was to make a small, but unique, contribution. Our aim in conceiving this conference was to build a bridge between top-level research and large-scale application of novel kinds of information security for mobile devices and communication. It has been a privilege to serve this event as a General Chair.

In this, first, year, MobiSec already attracted some excellent scientific papers, application studies, and contributions from industrial research from all over the world. Apart from that, the conference emphasis was on European leadership, by featuring a keynote from the European Commission, and contributions from the European Commssion-funded research projects SWIFT and OpenTC, which exhibited the strength of European industry and academia in the field. MobiSec's focus on knowledge transfer was supported by a tutorial on the Host Identification Protocol—a very topical contribution, which emphasizes the need to build in security in the autonomous, unsupervised parts of networking, in a cross-layer approach.

The papers at MobiSec 2009 dealt with a broad variety of subjects ranging from issues of trust in and security of mobile devices and embedded hardware security, over efficient cryptography for resource-restricted platforms, to advanced applications such as wireless sensor networks, user authentication, and privacy in an environment of autonomously communicating objects. With hindsight a leitmotif emerged from these contributions, which corroborated the idea behind MobiSec. A set of powerful tools have been created in various branches of the security discipline, which await combined application to build trust and security into mobile (that is, all future) networks, autonomous and personal devices, and pervasive applications.

Many people contributed to the success of MobiSec. It was a privilege to work with these dedicated persons, and I would like to thank them all for their efforts. The Organizing Committee as a whole created a frictionless, collaborative work atmosphere, which made my task an easy one. Antonio Lioy, who also did a lot of the local organization together with Daniele Mazzocchi, and Neeli Prasad did a great job assembling the Technical Program Committee and running the paper submission and review process. A high-quality conference cannot be created without the help of the distinguished members of the Program Committee—the soul of each scientific event. Shiguo Lian meticulously oversaw the preparation of the conference proceedings. The keynote speech by Bart van Caenegem of the EU Commission is very much appreciated, as well as the presentations by exponents of the EU research projects. Andrei Gurtov contributed a tutorial on an important and current topic. Special credit is due to Gergely Nagy, our conference coordinator from ICST, and his colleagues there, for their excellent support during the preparation of the event. It is hard to find an administrator under this workload who shows such responsiveness, competence, and pragmatic

attitude. Beatrix Ransburg and Eszter Hajdu oversaw the compilation of the CD ROM proceedings and this proceedings volume with calm and professionalism. Finally, I thank Imrich Chlamtac and the members of the Steering Committee for their support and guidance during these months and for entrusting me with the task of chairing MobiSec 2009.

Organizing a conference which is the first of its kind is always an adventure, if not venturous in the turbulent times we live in. We are delighted to have been part of it. Concluding, I am confident that all participants found MobiSec a stimulating and thought-provoking event, and enjoyed their time in Turin. We are looking forward to next year's MobiSec.

July 2009 Andreas U. Schmidt

Organization

Steering Committee Chair

Imrich Chlamtac — President, Create-Net Research Consortium, Trento, Italy

Steering Committee Members

Andreas U. Schmidt — Create-Net Research Consortium, Trento, Italy
Tibor Kovacs — Director of Business and Technology Affairs, ICST

General Chair

Andreas U. Schmidt — Create-Net Research Consortium, Trento, Italy

Technical Program Co-chairs

Neeli R. Prasad — Aalborg University, Denmark
Antonio Lioy — Politecnico di Torino, Italy

Local Arrangements Co-chairs

Daniele Mazzochi — Istituto Superiore Mario Boella (ISMB), Turin, Italy
Flaminia Luccio — University Ca'Foscari, Venice, Italy

Workshops Co-chairs

Rüdiger Grimm — University of Koblenz-Landau, Germany
Shiguo Lian — France Telecom R&D, Beijing, China

Panels Chair

Seung Woo Seo — Seoul National University, Korea

Publications Chair

Shiguo Lian — France Telecom R&D, Beijing, China

Web Chair

Giovanni Russello — Create-Net Research Consortium, Trento, Italy

Conference Coordinator

Gergely Nagy	ICST

Technical Program Committee

Selim Aissi	Intel, USA
Mahbubul Alam	Cisco, USA
Francesco Bergadano	Università degli Studi di Torino, Italy
Inhyok Cha	InterDigital Communications, USA
Rocky K. C. Chang	Hong Kong Polytechnic University, China
Hsiao-Hwa Chen	National Sun Yat-Sen University, Taiwan
Shin-Ming Chen	National Taiwan University, Taiwan
Tassos Dimitriou	AIT, Greece
Loic Duflot	SGDN/DCSSI, France
Ashutosh Dutta	Telcordia, USA
Stefanos Gritzalis	University of the Aegean, Greece
Markus Gueller	Infineon, Germany
Rajesh Gupta	University of California San Diego, USA
Marco Hauri	ASCOM, Switzerland
Mario Hoffmann	Fraunhofer SIT, Darmstadt, Germany
Jiankun Hu	RMIT University, Australia
Kazukuni Kobara	AIST, Japan
Geir Myrdahl Koien	Telenor, Norway
Shiguo Lian	France Telecom R&D, Beijing, China
Flaminia Luccio	Università Ca' Foscari, Venice, Italy
Fabio Martinelli	IIT and CNR, Pisa, Italy
Hassnaa Moustafa	France Telecom R&D (Orange Labs), France
Valtteri Niemi	Nokia, Finland
Vladimir Oleshchuk	UIA, Norway
Max Ott	NICTA, Australia
Jong Hyuk Park	Kyungnam University, Korea
Christos Politis	University of Kingston, UK
Anand Prasad	NEC Research Laboratories, Japan
Yi Qian	NIST, USA
Reijo Savola	VTT, Finland
Georg Schaathun	Surrey University, UK
Jean-Pierre Seifert	Samsung, USA
Yogendra Shah	InterDigital Communications
Chris Swan	Credit Suisse IT R&D, UK
Krzysztof Szczypiorski	Warsaw University of Technology, Poland
Allan Tomlinson	Royal Holloway University of London, UK
Janne Uusilehto	Nokia, Finland
Anna Vaccarelli	IIT and CNR, Pisa, Italy
Xin Wang	ContentGuard Inc., USA
Zheng Yan	Nokia Research Center, Finland

Table of Contents

On Trust Evaluation in Mobile Ad Hoc Networks

Dang Quan Nguyen[1], Louise Lamont[1], and Peter C. Mason[2]

[1] Communications Research Centre, Canada
{Dang.Nguyen,Louise.Lamont}@crc.gc.ca
[2] Defence Research & Development, Canada
Peter.Mason@drdc-rddc.gc.ca

Abstract. *Trust* has been considered as a social relationship between two individuals in human society. But, as computer science and networking have succeeded in using computers to automate many tasks, the concept of *trust* can be generalized to cover the reliability and relationships of non-human interaction, such as, for example, information gathering and data routing. This paper investigates the evaluation of trust in the context of ad hoc networks. Nodes evaluate each other's behaviour based on observables. A node then decides whether to trust another node to have certain innate abilities. We show how accurate such an evaluation could be. We also provide the minimum number of observations required to obtain an accurate evaluation, a result that indicates that observation-based trust in ad hoc networks will remain a challenging problem. The impact of making networking decisions using trust evaluation on the network connectivity is also examined. In this manner, quantitative decisions can be made concerning trust-based routing with the knowledge of the potential impact on connectivity.

Keywords: Ad hoc networks, security, trust evaluation, connectivity.

1 Introduction

A Mobile ad hoc network (MANET) consists of auto-configuring nodes that communicate with each other using wireless equipment. Such networks are infrastructureless, self-deploying and do not require a centralized entity. These advantages make MANETs suitable for critical uses: tactical military networks, disaster recovery, etc. Messages between two out-of-range nodes are routed in a multi-hop way, through intermediate nodes selected by MANET routing protocols (e.g. OLSR [1]).

These characteristics have a deep impact on security issues as they pose new challenges to the design of security solutions. One of these issues is concerned with trust management. The ad hoc environment is distributed and changing, meaning nodes can join and leave the network at any time. Therefore, traditional identification schemes based on a centralized authentication server are generally unsuitable for ad hoc networks. Ad hoc networks require new trust management designs to support a distributed environment and to be more robust against topology changes. One of the main components of trust management is *trust*

A.U. Schmidt and S. Lian (Eds.): MobiSec 2009, LNICST 17, pp. 1–13, 2009.

evaluation, or how to estimate the degree of trust between two nodes in an ad hoc network.

In this paper, we focus on the evaluation of trust. We define trust in the context of ad hoc networks and show how observations can be used to build an accurate estimate of trust. We also investigate the effect of decisions, based on trust evaluation, on one of the important properties of ad hoc networks which is connectivity. Indeed, trust decisions based on strict selection policy may result in few nodes selected, thus the network topology made of trusted nodes may be disconnected.

The remainder of this paper is organized as follows. In Section 2, we describe the existing work on trust computation. We start our study by giving a new definition of trust in ad hoc networks in Section 3 and discuss some aspects of this definition. Trust evaluation and estimation accuracy are explained in Section 4. We investigate the effect of trust evaluation on the network connectivity in Section 5. We conclude this paper in Section 6.

2 Related Work

There has been a significant amount of work done on trust. One of the earliest results on the topic within computer science [2] shows that trust can be formalized as a computational concept. Since then, research on trust has evolved in two main directions: *trust evaluation* and *trust sharing*.

Trust evaluation is concerned with the problem of estimating the trustworthiness of an entity (called a *node*) within a system, usually viewed as a network of interacting nodes. In [3], the authors propose a model for trust computation. This model defines trust as a subjective expectation a node has about another node based on the history of their encounters. This definition is probably closest to the one we will present. However, the work done in [3] is limited in the sense that it only takes into account a node's binary actions (cooperate or defect)– that is, the trust is discrete.

In [4], the authors use entropy to measure the uncertainty in trust relationship. This entropy is obtained from the probability that a node will perform some action. Such a probability is useful because it can be used as factor in predicting the behaviour of a node; that is, it can be used to estimate its trustworthiness. The authors do not, however, specify how this probability of node's compliance is arrived at in the first place.

On the other hand, trust sharing is concerned with the problem of sharing the estimation of a node's trustworthiness with other (usually distant) nodes and, conversely, of synthesizing all the received estimations. In [5], the authors propose an algorithm allowing indirect neighbours to estimate the trustworthiness of each other based on the trustworthiness of direct neighbours, as long as there exists a path between them. This algorithm of trust sharing treats it as a single real value between 0 and 1. It also assumes that trust propagation is multiplicative. Thus, given a node k which is a direct neighbour of nodes i and j, the level of trust node i puts in node j is the product of the trust values of node i in node

k and that of node k in node j. Trust sharing models such as this assume some transitivity of trust, but put stronger emphasis on information obtained from direct neighbours while attenuating trust values received via a multi-hop route. That is, local information carries a greater weight yet can still contribute to a global trust-sharing model.

In [6], the authors take a different approach to sharing trust values by considering trust as an opinion composed of a pair of real numbers (*trustvalue*, *confidence*) $\in [0,1] \times [0,1]$. While *trustvalue* is the estimation of the trustworthiness that node i puts in node j, *confidence* is the accuracy of the *trust value* assignment. In other words, *confidence* can be viewed as the quality of the estimation of trust. Therefore, when a node synthesizes opinions about a distant node, it must take into consideration the confidence value of each local *trustvalue*. We believe that this approach can give a more objective result of trust estimation than in [5] because it recognizes the subjectivity of each local trust estimation. This type of approach would lend itself well to a trust model that used a fuzzy logic reasoning engine.

In this paper, we consider the problem of trust evaluation and show how the quality of trust estimation can be quantified. To start, we state our definition of trust in the next Section.

3 Definition of Trust

Many existing definitions of trust are derivatives of authentication techniques which require encryption and a centralized authentication server. While authentication can provide a quick and efficient way to identify a node, implementing a practical and efficient authentication algorithm in an ad hoc environment remains an open problem [7]. We wish to decouple aspects of trust from authentication so that we may create an additional factor to be used as a tool in securing MANETs. One of the advantages of having a quantifiable and continuous value of trust available is that it allows flexibility in making certain security decisions so that trade-offs between security and functionality can be taken into consideration. We will return to this concept in Section 5.

We define the notion of trust of a given node in a MANET as the consistency of the node's behaviour. The behaviour is observed by other nodes in what is known as a watchdog approach [8]. A consequence of the watchdog approach is that it is observer-dependent; that is, an observed node can have different observational outcomes from the perspective of different neighbours. Let us define the *capacity* of a node to be the innate properties of that node. The node's behaviour will then be inherently tied to, and should reflect, its capacity. If a node behaves inconsistently, it is either because the node is being unfaithful to its capacity, in which case it is acting in an untrustworthy manner, or external factors (e.g. multipath and fading) are affecting its performance. In the latter case, we will assume that the observing nodes are also monitoring the environment and the link quality, can detect such factors, and compensate for them when making their observations.

Some examples of a node's behaviour related to security could be:

- The node's ability to reliably transmit periodic status updates that reflects parameters such as battery level, location and configuration.
- Forwarding packets in a timely manner based on management information base (MIB) bounded delay.
- The difference in the amount of data that should be forwarded and that is actually forwarded.

Our working definition of trust is meant to extract all aspects related to the capacity (as we have defined it) of a node from previous trust models. We do this for the purpose of allowing a quantitative measure of trust to be made which can then be used for making analytical decisions that affect network security. With this new definition of trust, nodes can proactively measure the trustworthiness of their neighbours through observation, without the need of challenging them. Moreover, the network does not need any centralized authentication server to assert signatures.

This definition of trust also allows us to decouple a node's capacity and its trustworthiness. For example, a node may have a large response delay because the throughput aspect of its capacity is low, but this node can still be trusted by other nodes as long as its response delays remain consistently large. As a result, for certain tasks nodes in the network could be selected as a function of both their trustworthiness and their capacities, e.g.: selecting highly capable nodes among those who are above a specified trust threshold.

On the assumption that nodes can monitor the behaviour of their neighbours using a variety of metrics, we will, for the rest of this paper, denote the outcome of a behaviour observation by X, a continuous random variable. X takes values between 0 and 1, thus the outcomes are normalized in the entire network. X is obtained by direct observation by a node i on a node j's behaviour. Values of X can be propagated to the other nodes in the network who can use them as they see fit. Therefore, a given node k can obtain many observation results of a distant node i from different sources (or observers).

The above assumption demands that the observers accurately report all observation results and the use of some cryptography mechanisms prevents the observation results from being modified while they are propagated in the network. The first assumption requires *objectivity* and the second *trust propagation*. These are strong assumptions and it is well recognized that both issues are themselves complex problems in MANET security that need to be addressed separately.

4 Accuracy of Trust Evaluation

In this Section, we are interested in trust as a measure of the consistency of a node's behaviour as objectively observed by one or many different observers.

4.1 Estimation of a Node's Capacities

Let $X_1, X_2, \ldots, X_n$ be different observation outcomes of a node i reported by different sources to a node j. If node j computes a weighted average of these values to

estimate the capacity of i, then our main concern is how accurate this estimation would be when compared to node i's true capacity. We can subsequently quantify the number of observations needed by node j in order to achieve an acceptable level of confidence in its estimation of node i's capacity. Node j can then decide whether it should trust node i to have such an estimated capacity.

Let $Y = c_1X_1 + c_2X_2 + \ldots + c_nX_n$ be an estimation of node i's capacity, with $0 \leqslant c_i \leqslant 1$ and $\sum_{i=1}^{n} c_i = 1$. We introduce the weights c_i to allow node j to assign different importance to the values X_i, for example: recent observations are more important than old ones. All random variables X_i are assumed to be independent and identically distributed.

Since X measures the observational outcomes of a node's behaviour, $\mu = E[X]$ represents the capacity of this node. Our problem can be formulated as follows: given $\delta \geqslant 0$ and $\epsilon \geqslant 0$, what is the probability that an estimation Y of μ can achieve an accuracy of δ, and conversely, how many observations (n) are needed in order to achieve an estimation of accuracy δ with the probability $1 - \epsilon$?

Theorem 1 (Chernoff bound). *Let* $\mu = E[X]$. *Denote by* $\phi_X(s) = \int_{-\infty}^{\infty} \mathrm{e}^{sx} f_X(x)dx$ *the moment generating function of* X. *We have*

$$P\left[|Y - \mu| \geqslant \delta\right] \leqslant \min_{s \geqslant 0} \left(\mathrm{e}^{-s(\delta+\mu)} \prod_{i=1}^{n} \phi_X(c_i s) \right) + \min_{s \geqslant 0} \left(\mathrm{e}^{-s(\delta-\mu)} \prod_{i=1}^{n} \phi_X(-c_i s) \right)$$

Proof. We have

$$P\left[|Y - \mu| \geqslant \delta\right] = P\left[Y - \mu \geqslant \delta\right] + P\left[Y - \mu \leqslant -\delta\right].$$

Apply Chernoff bound to random variable Y in the first term yields

$$P\left[Y - \mu \geqslant \delta\right] = P\left[Y \geqslant \delta + \mu\right] \leqslant \min_{s \geqslant 0} \left(\mathrm{e}^{-s(\delta+\mu)} \phi_Y(s) \right).$$

Since X_i are independent and identically distributed random variables:

$$\phi_Y(s) = \phi_{\sum_{i=1}^{n} c_i X_i}(s) = \prod_{i=1}^{n} \phi_{c_i X_i}(s) = \prod_{i=1}^{n} \phi_X(c_i s).$$

And hence

$$P\left[Y - \mu \geqslant \delta\right] \leqslant \min_{s \geqslant 0} \left(\mathrm{e}^{-s(\delta+\mu)} \prod_{i=1}^{n} \phi_X(c_i s) \right).$$

Similarly, applying the Chernoff bound to the random variable $Z = -Y$ in the second term yields

$$P\left[Y - \mu \leqslant -\delta\right] \leqslant \min_{s \geqslant 0} \left(\mathrm{e}^{-s(\delta-\mu)} \prod_{i=1}^{n} \phi_X(-c_i s) \right)$$

which ends the proof. □

If X is a gaussian random variable with mean μ and variance σ^2, then we have the following result.

Corollary 1. *Let* $\xi = \sum_{i=1}^{n} c_i^2$. *If* $X \sim \mathrm{G}(\mu, \sigma^2)$, *then*

$$P\left[|Y - \mu| \geqslant \delta\right] \leqslant 2\exp\left(-\frac{\delta^2}{2\sigma^2\xi}\right)$$

Proof. If $X \sim \mathrm{G}(\mu, \sigma^2)$ then $\phi_X(s) = \exp\left(\mu s + \frac{\sigma^2 s^2}{2}\right)$.

Therefore

$$\begin{aligned}\prod_{i=1}^{n} \phi_X(c_i s) &= \prod_{i=1}^{n} \exp\left(\mu c_i s + \frac{\sigma^2 c_i^2 s^2}{2}\right) \\ &= \exp\left(\mu \textstyle\sum_{i=1}^{n} c_i s + \frac{\sigma^2 \sum_{i=1}^{n} c_i^2 s^2}{2}\right) \\ &= \exp\left(\mu s + \frac{\sigma^2 \xi s^2}{2}\right).\end{aligned}$$

And hence

$$\min_{s \geqslant 0}\left(\mathrm{e}^{-s(\delta+\mu)} \prod_{i=1}^{n} \phi_X(c_i s)\right) = \min_{s \geqslant 0}\left(\exp\left(-\delta s + \frac{\sigma^2 \xi s^2}{2}\right)\right).$$

The expression $-\delta s + \frac{\sigma^2 \xi s^2}{2}$ has a minimum value of $-\frac{\delta^2}{2\sigma^2\xi}$ when $s = \frac{\delta}{\sigma^2 \xi} \geqslant 0$.

Thus

$$\min_{s \geqslant 0}\left(\mathrm{e}^{-s(\delta+\mu)} \prod_{i=1}^{n} \phi_X(c_i s)\right) = \exp\left(-\frac{\delta^2}{2\sigma^2\xi}\right).$$

Similar calculations give

$$\min_{s \geqslant 0}\left(\mathrm{e}^{-s(\delta-\mu)} \prod_{i=1}^{n} \phi_X(-c_i s)\right) = \exp\left(-\frac{\delta^2}{2\sigma^2\xi}\right).$$

The assertion thus follows from Theorem 1. □

An interesting conclusion we can draw from Corollary 1 is that the accuracy of the Y-estimation does not depend on the true capacity μ of the subject node. That is, in other words, if node j has received n observations $X_1, \ldots, X_n$ of node i, then it can estimate the capacity of node i with a certain degree of confidence, even if this estimation indicates that node i's capacity is low. Conversely, when some early-arriving reports indicate that node i's capacity is rather high, node j should not rely entirely on this small number of observations to conclude this with a high degree of confidence. Since this result shows that the capacity μ of a node and the estimation accuracy (represented by δ) are statistically unrelated, we do not need to have *a priori* knowledge of a node's capacity in order to draw conclusions about its trustworthiness.

4.2 Number of Observations Required

The following corollary gives a lower bound on the number of observations needed in order to achieve a desired degree of confidence in the estimation.

Corollary 2. *Suppose that* $X \sim \mathrm{G}(\mu, \sigma^2)$. *Given* $0 < \epsilon < 1$, *the minimum number of observations needed in order to achieve an estimation of accuracy* δ *with the probability* $1 - \epsilon$ *is*

$$n \geqslant \frac{2\sigma^2}{\delta^2} \ln\left(\frac{2}{\epsilon}\right)$$

Proof. The inequality in Corollary 1 can be rewritten as

$$P\left[|Y - \mu| \geqslant \delta\right] \leqslant \min_{\xi}\left(2\exp\left(-\frac{\delta^2}{2\sigma^2\xi}\right)\right)$$

where $\xi = \sum_{i=1}^{n} c_i^2 \geqslant \frac{1}{n}$ by Cauchy-Schwartz inequality. Equality occurs when $c_1 = \ldots = c_n = \frac{1}{n}$.

Therefore

$$P\left[|Y - \mu| \geqslant \delta\right] \leqslant 2\exp\left(-\frac{\delta^2 n}{2\sigma^2}\right).$$

And hence

$$P\left[|Y - \mu| < \delta\right] \geqslant 1 - 2\exp\left(-\frac{\delta^2 n}{2\sigma^2}\right) \geqslant 1 - \epsilon$$

yields the desired result. □

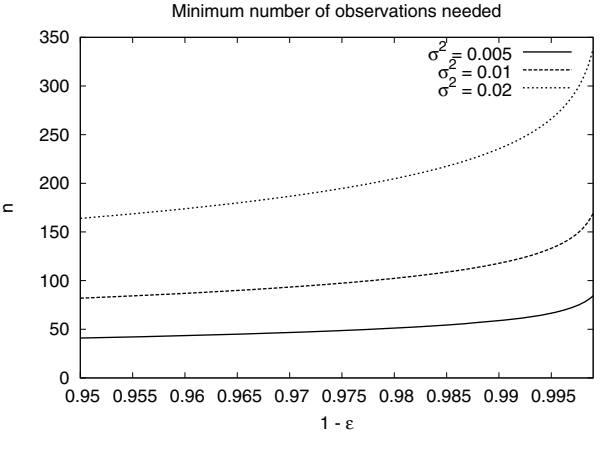


Fig. 1. Minimum number of observations needed to satisfy probability $1 - \epsilon$, with $\delta = 0.03$

Figure 1 shows the minimum number of observations needed in order to achieve an estimation having an accuracy of ± 0.03 ($\delta = 0.03$) with probability at least 95%, for three different values of the variance, σ^2.

Corollary 2 shows that the minimum number of observations required is proportional to the maliciousness of a node which is represented by the standard deviation σ. The deviation measures the lack of consistency in a node's behaviour and we assume intentionally inconsistent behaviour is malicious or untrustworthy. The more inconsistently a node behaves, the more observations we need in order to accurately estimate its capacity. This implies that a greater number of observations are required in order to identify untrustworthy nodes, a fact that makes doing so a more onerous task.

Corollary 2 also gives a lower bound on the number of observations needed to perform an accurate estimation. This lower bound is obtained when equal weights are assigned to each of the observations. Equal weighting of observations is an unlikely scenario: it is more likely a node will grant more importance to recent observations than to stale ones, or more importance to its own observations than to the ones reported by the other nodes. Therefore, the minimum number of observations needed will be higher but will still be accurately quantifiable.

An additional parameter of interest that can be extracted from our calculations is the number of observations required to achieve different accuracies in trust assessment of a node. Figure 2 shows the probability of having an estimation achieve an accuracy δ as a function of the number of observations for a given variance. This figure shows that achieving a very high level of confidence in an assessment comes at great cost with respect to the number of observations required. In this example, it only requires approximately twenty observations to achieve an accuracy of within 4% ($\delta = 0.04$) with 80% confidence but doubling the accuracy to 2% at the same confidence level increases the number of required observations by a factor of six, to $n = 120$. Again, a goal of this work is to allow

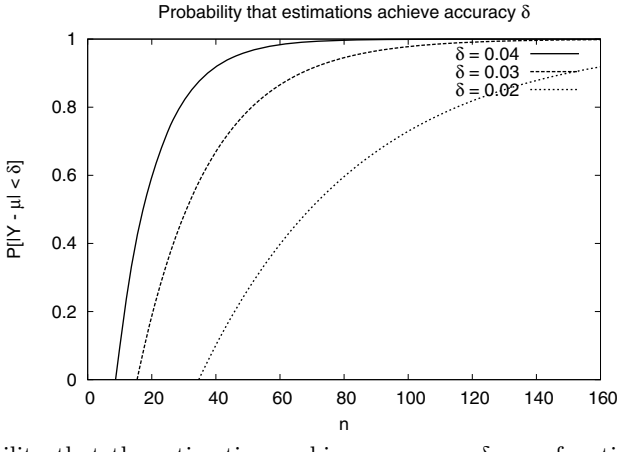


Fig. 2. Probability that the estimations achieve accuracy δ as a function of n, with $\sigma^2 = 0.01$

these tradeoffs to be understood so that decisions using trust as a parameter can be weighed appropriately.

5 Trust and Network Connectivity

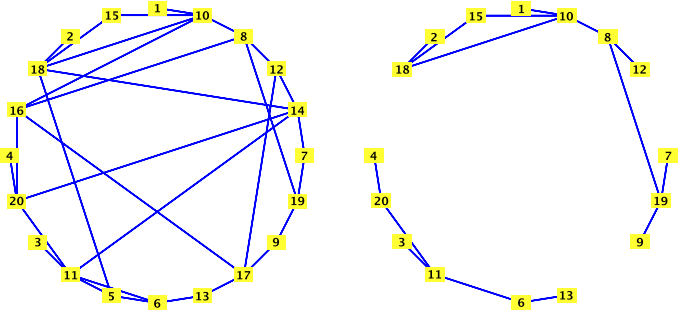
In this section, we study an example where decisions based on trust may have an effect on the connectivity of the network. In particular, we are interested in the probability of the network graph remaining connected if some nodes of the network are untrusted and thus either do not, or are not permitted to, participate in routing. This probability of connectivity is useful for the configuration of trust-based routing. Indeed, when a node extracts a trust topology out of the network graph by excluding nodes having insufficient trustworthiness, it may obtain a partitioned graph and hence trust-based routing may not be available to all destinations.

5.1 Connectivity of Trust-Based Networks

Network connectivity is an important issue in networking and distributed systems. Research on this topic ranges from graph theory [9,10,12] to physics-related domains such as percolation theory [11]. In [12], the authors show that if n nodes of a network are placed uniformly and independently in a unit disc, then the network is connected with a probability asymptotically tending to 1 if and only if each node has $\log n + c(n)$ neighbours and $c(n) \to \infty$ as $n \to \infty$.

In this section, we consider a connected random graph G characterized by n nodes and average density d (or number of neighbours per node). We derive an upper-bound of the connectivity probability for this graph when a subset S, $0 \leqslant |S| \leqslant n$, of randomly chosen nodes in G is untrusted and removed from G.

Figure 3(a) shows an example of a random graph having 20 nodes and average density 3. Nodes {5, 14, 16, 17}, randomly chosen, are untrusted and removed



(a) Initial network graph G

(b) Trust topology, nodes {5,14,16,17} are untrusted and removed from G

Fig. 3. A random graph composed of 20 nodes with 3 neighbours in average per node

from the original graph to obtain a trust topology of the network. This topology is partitioned into two components (see Figure 3(b)).

The following theorem gives us an upper bound of the probability that the trust topology remains connected.

Theorem 2 (Trust-based connectivity).
Let $\rho = \frac{|S|}{n}$*, the probability that* $G\backslash S$ *is connected is*

$$P\left[G\backslash S \text{ is connected}\right] \leqslant 1 - \left(\rho\left(2-\rho\right)\right)^{\frac{1}{2}d(1-\rho)}$$

Proof. We prove this theorem by first quantifying the number of edges that are removed from G due to the removal of nodes in S. Then, the remaining induced subgraph $G\backslash S$ is connected if and only if it still has at least one spanning tree.

Since G has n nodes and d neighbours per node on average, the probability that there exists a link between any two nodes is $\frac{d}{n}$. Therefore, the expected number of induced edges of $G\backslash S$, which has $(1-\rho)n$ nodes, is

$$\|G\backslash S\| = \frac{1}{2}\left((1-\rho)n\right)^2 \frac{d}{n} = \|G\|(1-\rho)^2.$$

Hence, the expected number of edges removed from G is

$$\begin{aligned}\|G\| - \|G\backslash S\| &= \|G\|\left(1-(1-\rho)^2\right)\\ &= \|G\|\rho(2-\rho)\end{aligned}$$

which means an edge of G is arbitrarily removed with probability $\rho(2-\rho)$.

Let k be the number of edge-disjoint spanning trees in $G\backslash S$. As each spanning tree in $G\backslash S$ has $(1-\rho)n-1$ edges, we have

$$k \leqslant k_{max} = \frac{\|G\|(1-\rho)^2}{(1-\rho)n-1} \approx \frac{1}{2}d(1-\rho).$$

$G\backslash S$ is disconnected if and only if all its k edge-disjoint spanning trees are disconnected, i.e. at least k edges must be removed from $G\backslash S$ to disconnect it. Hence

$$P\left[G\backslash S \text{ is disconnected}\right] \geqslant \left(\rho(2-\rho)\right)^k \geqslant \left(\rho(2-\rho)\right)^{\frac{1}{2}d(1-\rho)}$$

which ends the proof. □

5.2 Validation

We validate the above analysis by simulation. To start, we fix a value of ρ increasing from 0 to 0.95 by step 0.05, i.e. $\rho = 0, 0.05, 0.1, \ldots, 0.95$. For each value of ρ, we generate 10,000 random graphs. Each graph has 100 nodes and average density $\log_2(100)+1$. Therefore, we ensure that most of the initial graphs are connected (see [12]).

For each random graph G, $\lfloor \rho n \rfloor$ nodes are removed from G. The edges incident to these nodes are also removed. We calculate the percentage of graphs that

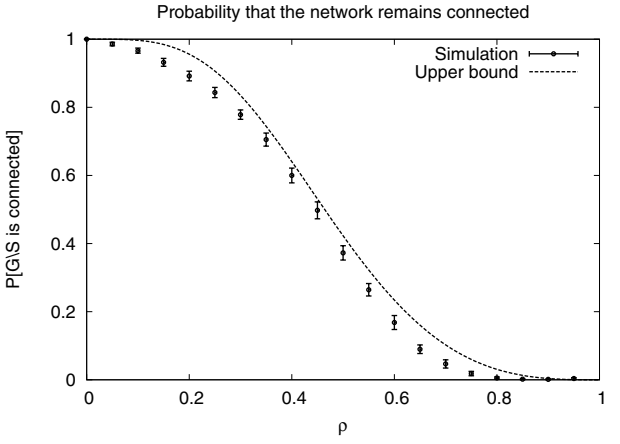


Fig. 4. Probability that a 100-node network ($d = \log_2(100) + 1$) remains connected in presence of a subset S of untrusted nodes, with $|S| = \rho n$

remain connected along with the standard deviation. In total, 200,000 random graphs are generated. The simulations are done using Maple software ([13]).

Figure 4 compares simulation results to analysis. We see that the probabilities of connectivity obtained by simulations closely follow the trend of the upper-bound probabilities obtained by analysis.

The results shown in Fig. 4 demonstrate the potential impact of using strict policies on trust to implement concepts like trust-based routing. If the threshold of required trust is set too high, there is a strong likelihood that a critical number of nodes will be excluded from the network, endangering connectivity. Knowledge of this relationship between trust-level and potential network segregation will allow security decisions to be made in which assuming different levels of risk (routing through less-trusted nodes) can be balanced against the value of increasing the probability of successful message transmission.

6 Conclusion

In this paper, we investigate the issue of trust evaluation and estimation accuracy for ad hoc networks. We start our study by giving a clear definition of trust in the context of ad hoc networks. This definition extracts the physically observable aspects of a nodes behaviour so that each node in the network can decide whether it can trust another node to have certain capacities. We then show that a node's true capacity and its estimation accuracy are statistically independent, given that a node's behaviour follows a normal distribution law. We also provide a minimum number of observations required in order to obtain an accurate estimation of a node's capacity. Given that this minimum number is large, we have shown that an implementation of an analytical trust model will require either a large number of independent observations done in parallel or

the ability to cache and safely propagate observation information through the network.

A motivation of this work is to quantify the trade-offs and requirements that will naturally arise by defining trust in this manner for ad hoc networks. To that end, we present an example showing what effect trust-based decisions may have on network connectivity. We derive an upper-bound probability of the network remaining connected when some nodes in that network are untrusted. This information could be used so that trust-based routing is available to as many nodes in the network as possible while simultaneously having an understanding of the measure of risk that is being assumed to do so.

In future work, we can study different mobility scenarios (e.g. time required to compute observations versus speed of nodes) as additional parameters to better understand the tradeoffs and practicability of using trust for security decisions. In addition, an examination of the avenues of attack on this trust model can be considered along with suggestions for mitigating their effects.

Acknowledgement

This work is funded by Defence Research & Development Canada (DRDC).

References

1. Adjih, C., Clausen, T., Jacquet, P., Laouiti, A., Minet, P., Muhlethaler, P., Qayyum, A., Viennot, L.: Optimized Link State Routing Protocol. RFC 3626, IETF (October 2003)
2. Marsh, S.P.: Formalising trust as a computational concept. PhD thesis, University of Stirling (1994)
3. Mui, L., Mohtashemi, M., Halberstadt, A.: A Computational Model of Trust and Reputation. In: Proc. of 35^{th} Hawaii International Conference on System Sciences, HICSS 2002, Hawaii, USA (January 2002)
4. Sun, Y., Yu, W., Han, Z., Ray Liu, K.J.: Trust Modeling and Evaluation in Ad Hoc Networks. In: Proc. of IEEE Global Telecommunications Conference, GLOBECOM'05, St, Louis MO, USA (December 2005)
5. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of Trust and Distrust. In: Proc. of 13^{th} International Conference on World Wide Web, WWW 2004, New York NY, USA (May 2004)
6. Theodorakopoulos, G., Baras, J.S.: Trust Evaluation in Ad-Hoc Networks. In: Proc. of 3^{rd} ACM Workshop on Wireless Security, WiSe 2004, Philadelphia PA, USA (October 2004)
7. Tang, H., Salmanian, M.: Lightweight Integrated Authentication Protocol for Tactical MANETs. In: Proc. of IEEE TrustCom 2008, Zhangjiajie, China (November 2008)
8. Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In: Proc. of the Sixth Annual Conference on Mobile Computing and Networking, Boston, MA, USA (August 2000)
9. Gilbert, E.N.: Random Plane Networks. Journal of the Society for Industrial and Applied Mathematics 9(4), 533–543 (1961)

10. Philips, T., Panwar, S., Tantawi, A.: Connectivity Properties of a Packet Radio Network Model. IEEE Transactions on Information Theory 35(5), 1044–1047 (1989)
11. Meester, R., Roy, R.: Continuum Percolation. Cambridge University Press, Cambridge (1996)
12. Gupta, P., Kumar, P.R.: Critical Power for Asymptotic Connectivity in Wireless Networks. In: McEneany, W.M., et al. (eds.) Stochastic Analysis, Control, Optimization and Applications, pp. 547–566. Birkhauser, Boston (1998)
13. Maplesoft: Math Software for Engineers, Educators & Students, http://www.maplesoft.com

A Distributed Data Storage Scheme for Sensor Networks

Abhishek Parakh and Subhash Kak

Computer Science Department
Oklahoma State University, Stillwater OK 74075
{parakh,subhashk}@cs.okstate.edu

Abstract. We present a data storage scheme for sensor networks that achieves the targets of encryption and distributed storage simultaneously. We partition the data to be stored into numerous pieces such that at least a specific number of them have to be brought together to recreate the data. The procedure for creation of partitions does not use any encryption key and the pieces are implicitly secure. These pieces are then distributed over random sensors for storage. Capture or malfunction of one or more (less than a threshold number of sensors) does not compromise the data. The scheme provides protection against compromise of data in specific sensors due to physical capture or malfunction.

Keywords: Distributed data storage, sensor networks.

1 Introduction

Sensors may deployed in hostile environment where they may be prone to dangers ranging from environmental hazards to physical capture. Under such circumstances the data and encryption keys stored on these sensors is vulnerable to compromise.

A number of techniques have been proposed for secure communication between sensors using key management and data encryption [1,2,3,4]. Some techniques aim at secure routing [5,6], intrusion detection [7] and others describe a mechanism for moving sensitive data around in the network from time to time [8]. A few researchers propose distributed data storage [15,16,17] but do not adequately address the question of security or use explicit encryption techniques to secure data which leaves the question of secure storage of encryption/decryption keys unanswered.

To go beyond the present approaches, one may incorporate further security within the system by using another layer that increases the space that the intruder must search in order to break a cipher [9,10]. Here, we propose an implicitly secure data partitioning scheme whose security is distributed amongst many sensors. In contrast to hash-based distributed security models for wireless sensor networks [18,19], we consider a more general method of data partitioning. In this approach, stored data is partitioned into two or more pieces and stored at randomly chosen sensors on the network. In scenarios where one or more

A.U. Schmidt and S. Lian (Eds.): MobiSec 2009, LNICST 17, pp. 14–22, 2009.

pieces may be at the danger of being lost or inaccessible due to sensor failure or capture, one may employ schemes that can recreate the data from a subset of original pieces.

A number of schemes have been proposed in the communications context for splitting and sharing of decryption keys [11,20]. These schemes fall under the category of "secret sharing schemes", where the decryption key is considered to be a secret. Motivated by the need to have an analog of the case where several officers must simultaneously use their keys before a bank vault or a safe deposit box can be opened, these schemes do not consider the requirement of data protection for a single party. Further, in any secret sharing scheme it is assumed that the encrypted data is stored in a secure place and that none of it can be compromised without the decryption key.

In this paper, we protect data by distributing its parts over various sensors. The idea of making these partitions is a generalization of the use of 3 or 9 roots of a number in a cubic transformation [12]. The scheme we present is simple and easily implementable.

We would like to stress that the presented scheme is different from Shamir's secret sharing scheme which takes the advantage of polynomial interpolation. Further, Shamir's scheme maps the secret as points on the y-axis,, whereas the scheme proposed in this paper maps the secret as points on the x-axis, as roots of a polynomial.

2 Proposed Data Partitioning Scheme

By the fundamental theorem of algebra, every equation of k^{th} degree has k roots. We use this fact to partition data into k partitions such that each of the partition is stored on a different sensor. No explicit encryption of data is required to secure each partition. The partitions in themselves do not reveal any information and hence are implicitly secure. Only when all the partitions are brought together is the data revealed.

Consider an equation of degree k

$$x^k + a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + ... + a_1x + a_0 = 0 \tag{1}$$

Equation 1 has k roots denoted by $\{r_1, r_2, ..., r_k\} \subseteq \{\text{set of complex numbers}\}$ and can be rewritten as

$$(x - r_1)(x - r_2)...(x - r_k) = 0 \tag{2}$$

In cryptography, it is more convenient to use the finite field $\mathbb{Z}_p$ where p is a large prime. If we replace a_0 in (1) with the data $d \in \mathbb{Z}_p$ that we wish to partition then,

$$x^k + \sum_{i=1}^{k-1} a_{k-i}x^{k-i} + d \equiv 0 \bmod p \tag{3}$$

where $0 \leq a_i \leq p-1$ and $0 \leq d \leq p-1$. (Note that one may alternatively use $-d$ in (3) instead of d.) This may be rewritten as

$$\prod_{i=1}^{k}(x - r_i) \equiv 0 \bmod p \tag{4}$$

where $1 \leq r_i \leq p-1$. The roots, r_i, are the partitions. It is clear that the term d in (3) is independent of variable x and therefore

$$\prod_{i=1}^{k} r_i \equiv d \bmod p \tag{5}$$

If we allow the coefficients in (3) to take values $a_1 = a_2 = ... = a_{k-1} = 0$, then (3) will have k roots only if $GCD(p-1, k) \neq 1$ and $\exists b \in \mathbb{Z}_p$ such that d is the k^{th} power of b. One simple way to chose such a p would be to choose a prime of the form $(k \cdot s + 1)$, where $s \in \mathbb{N}$. However, such a choice would not provide good security because knowledge of the number of roots and one of the partitions would be sufficient to recreate the original data by computing the k^{th} power of that partition. Furthermore, not all values of d will have a k^{th} root and hence one cannot use any arbitrary integer, which would typically be required. Therefore, one of the restrictions on choosing the coefficients is that not all of them are simultaneously zero.

For example, if the data needs to be divided into two parts then an equation of second degree is chosen and the roots computed. If we represent this general equation by

$$x^2 + a_1 x + d \equiv 0 \bmod p \tag{6}$$

then the two roots can be calculated by solving the following equation modulo p,

$$x = \frac{-a_1 \pm \sqrt{a_1^2 - 4d}}{2} \tag{7}$$

which has a solution in $\mathbb{Z}_p$ only if the square root $\sqrt{a_1^2 - 4d}$ exists modulo p. If the square root does not exist then a different value of a_1 needs to be chosen. We present a practical way of choosing the coefficients below. However, this brings out the second restriction on the coefficients, i.e. they should be so chosen such that a solution to the equation exists in $\mathbb{Z}_p$.

Theorem 1. *If the coefficients a_i, $1 \leq i \leq k-1$ in equation (3) are not all simultaneously zero, are chosen randomly and uniformly from the field, then the knowledge of any $k-1$ roots of the equation, such that equation (4) holds, does not provide any information about the value of d with a probability greater than that of a random guess of $1/p$.*

Proof. Given a specific d, the coefficients in (3) can be chosen to satisfy (4) in $\mathbb{Z}_p$ in the following manner. Choose at randomly and uniformly from the field

$k-1$ random roots $r_1, r_2, ..., r_{k-1}$. Then k^{th} root r_k can be computed by solving the following equation,

$$r_k = d \cdot (r_1 \cdot r_2 \cdot ... \cdot r_{k-1})^{-1} \bmod p \tag{8}$$

Since the roots are randomly chosen from a uniform distribution in $\mathbb{Z}_p$, the probability of guessing r_k without knowing the value of d is $1/p$. Conversely, d cannot be estimated with a probability greater than $1/p$ without knowing the k^{th} root r_k. □

It follows from Theorem 1 that data is represented as a multiple of k numbers in the finite field.

Example 1. Let data $d = 10$, prime $p = 31$, and let $k = 3$. We need to partition the data into three parts for which we will need to use a cubic equation, $x^3 + a_2x^2 + a_1x - d \equiv 0 \bmod p$.

We can find the equation satisfying the required properties using Theorem 1. Assume, $(x-r_1)(x-r_2)(x-r_3) \equiv 0 \bmod 31$. We randomly choose 2 roots from the field, $r_1 = 19$ and $r_2 = 22$. Therefore, $r_3 \equiv d\cdot(r_1\cdot r_2)^{-1} \equiv 10\cdot(19\cdot 22)^{-1} \bmod 31 \equiv 11$. The equation becomes $(x-r_1)(x-r_2)(x-r_3) \equiv x^3-21x^2+x-10 \equiv 0 \bmod 31$, where the coefficients are $a_1 = 1$ and $a_2 = -21$ and the partitions are 11, 22 and 19.

Choosing the Coefficients: We described above two conditions that must be satisfied by the coefficients. The first condition was that not all the coefficients are simultaneously zero and second that the choice of coefficients should result in an equation with roots in $\mathbb{Z}_p$. Since no generalized method for solving equations of degree higher than 4 exists [21], a numerical method must be used which becomes impractical as the number of partitions grows. An easier method to compute the coefficients is exemplified by Theorem 1 and Example 1.

One might ask why should we want to compute the coefficients if we already have all the roots? We answer this question a little later.

Introducing Redundancy: In situations when the data pieces stored on one or more (less than a threshold number of) sensors over the sensor network may not be accessible, then other sensors should be able to collaborate to recreate the data from the available pieces. The procedure outlined below extends the k partitions to n partitions such that only k of them need to brought together to recreate the data. If $\{r_1, r_2, \ldots, r_k\}$ is the original set of partitions then they can be mapped into a set of n partitions $\{p_1, p_2, \ldots, p_n\}$ by the use of a mapping function based on linear algebra. If we construct n linearly independent equations such that

$$\begin{aligned} a_{11}r_1 + a_{12}r_2 + \ldots + a_{1k}r_k &= c_1 \\ a_{21}r_1 + a_{22}r_2 + \ldots + a_{2k}r_k &= c_2 \\ &\vdots \\ a_{n1}r_1 + a_{n2}r_2 + \ldots + a_{nk}r_k &= c_n \end{aligned}$$

where numbers a_{ij} are randomly and uniformly chosen from the finite field $\mathbb{Z}_p$, then the n new partitions are $p_i = \{a_{i1}, a_{i2}, \ldots, a_{ik},\ c_i\}$, $1 \leq i \leq n$. The above linear equation can be written as matrix operation,

$$\begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1k} \\ a_{21} & a_{22} & \ldots & a_{2k} \\ \vdots & & & \\ a_{n1} & a_{n2} & \ldots & a_{nk} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_k \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$$

To recreate r_j, $1 \leq j \leq k$ from the new partitions, any k of them can be brought together,

$$\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_k \end{bmatrix} = \begin{bmatrix} a_{m1} & a_{m2} & \ldots & a_{mk} \\ a_{n1} & a_{n2} & \ldots & a_{nk} \\ \vdots & & & \\ a_{i1} & a_{i2} & \ldots & a_{ik} \end{bmatrix}^{-1}_{k \times k} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix}_{k \times 1}$$

A feature of the presented scheme is that new partitions may be added and deleted without affecting any of the existing partitions.

An alternate approach to partitioning: Once a sensor has computed all the k roots then it may compute the equation resulting from (4) and store one or all of the roots on different sensors and the coefficients on different sensors. Recreation of original data can now be performed in two ways: either using (5) or choosing one of the roots at random and retrieving the coefficients and substituting the appropriate values in the equation (3) to compute

$$-a_0 \equiv x^k + a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + ... + a_1 x \bmod p \tag{9}$$

Therefore, the recreated data $d \equiv (p - a_0) \bmod p$. Parenthetically, this may provide a scheme for fault tolerance and partition verification. Additionally, a sensor may store just one of the roots and $k - 1$ coefficients on the network. These together represent k partitions.

Note. Distinct sets of coefficients (for a given constant term in the equation) result in distinct sets of roots and vice versa. This is because two distinct sets of coefficients represent distinct polynomials because two polynomials are said to be equal if and only if they have the same coefficients. By the fundamental theorem of algebra, every polynomial has unique set of roots.

Two sets of roots R_1 and R_2 are distinct if and only if $\exists r_i \forall r_j (r_i \neq r_j)$, where $r_i \in R_1$ and $r_j \in R_2$. To compute the corresponding polynomials and the two sets coefficients C_1 and C_2, we perform $\prod_{i=1, r_i \in R_1}^{k} (x - r_i) \equiv 0 \bmod p$ and $\prod_{j=1, r_j \in R_2}^{k} (x - r_j) \equiv 0 \bmod p$ and read the coefficients from the resulting polynomials, respectively. It is clear the at least one of the factors of the two polynomials is distinct because at least one of the roots is distinct; hence the resulting polynomials for a distinct set of roots are distinct.

Theorem 2. *Determining the coefficients of a polynomial of degree $k \geq 2$ in a finite field $\mathbb{Z}_p$, where p is prime, by brute force, requires $\Omega(\lceil \frac{p^{k-1}}{(k-1)!} \rceil)$ computations.*

Proof. If A represents the set of coefficients $A = \{a_1, a_2, ..., a_{k-1}\}$, where $0 \leq a_i \leq p-1$, then by the above note each distinct instance of set A gives rise to a distinct set of roots $R = \{r_1, r_2, ..., r_k\}$, and, conversely, every distinct instance of set R gives rise to a distinct set of coefficients A. Therefore, if we fix d to a constant, then (5) can be used to compute the set of roots. Every distinct set of roots is therefore a $k-1$ combination of a multiset [13,14], where each element has infinite multiplicity, and equivalently a $k-1$ combination of set $S = \{0, 1, 2, ..., p-1\}$ with repetition allowed. Thus, the number of possibilities for the choices of coefficients is given by the following expression

$$\begin{aligned} \left\langle \begin{matrix} p \\ k-1 \end{matrix} \right\rangle &= \binom{p-1+(k-1)}{k-1} \\ &= \tfrac{(p+k-2)!}{(p-1)!(k-1)!} \\ &= \tfrac{(p+k-2)(p+k-3)...(p+k-k)(p-1)!}{(p-1)!(k-1)!} \\ &= \tfrac{(p+k-2)(p+k-3)...p}{(k-1)!} \\ &\geq \lceil \tfrac{p^{k-1}}{(k-1)!} \rceil \end{aligned} \tag{10}$$

Here we have used the fact that in practice $p \gg k \geq 2$, hence the result. We have ignored the one prohibited case of all coefficients being zero, which has no effect on our result. □

3 A Stronger Variation to the Protocol Modulo a Composite Number

An additional layer of security may be added to the implementation by performing computations modulo a composite number $n = p \cdot q$, p and q are primes, and using an encryption exponent to encrypt the data before computing the roots of the equation. In such a variation, knowledge of all the roots and coefficients of the equation will not reveal any information about the data and the adversary will require to know the secret factors of n. For this we can use an equation such as the one below:

$$x^k + \sum_{i=1}^{k-1} a_{k-i} x^{k-i} + d^y \equiv 0 \bmod n \tag{11}$$

where y is a secretly chosen exponent and $GCD(y, n) = 1$. If the coefficients are chosen such that (11) has k roots then

$$\prod_{i=1}^{k} r_i \equiv d^y \bmod n \tag{12}$$

Appropriate coefficients may be chosen in a manner similar to that described in previous sections. It is clear that compromise of all the roots and coefficients

will at the most reveal $c = d^y \bmod n$. In order to compute the original value of d, the adversary will require the factors of n which are held secret by the sensor which owns the data.

4 Addressing the Data Partitions

The previous sections consider the security of the proposed scheme when prime p and composite n are public knowledge. However, there is nothing that compels the user to disclose the values of p and n. If we assume that p and n are secret values, then the partitions may be stored in the form of an "encrypted link list", which is a list in which every pointer is in encrypted form and in order to find out which node the present node points to, a party needs to decrypt the pointer which can be done only if certain secret information is known.

If we assume that p and n are public values then the pointer can be so encrypted that each decryption either leads to multiple addresses or depends on the knowledge of the factors or both. Only the legitimate party will know which of the multiple addresses is to be picked.

Alternatively, one may use a random number generator and generate a random sequence of sensor IDs using a secret seed. One way to generate this seed may be to find the hash of the original data and use it to seed the generated sequence and keep the hash secret.

5 Future Work and Other Applications

Our approach leads to interesting research issues such as the optimal way to distribute the data pieces in a network of n sensors. Also in the case when the sensors are moving, one needs to investigate as to how the partitions need to be reallocated so that the original sensor is always able to access the pieces when needed. Further, questions of load balancing so that no one sensor is storing a very large number of partitions and the partitions are as evenly distributed over the network as possible, needs to be investigated.

Yet another application of the presented scheme is in Internet voting protocols. Internet voting is a challenge for cryptography because of its opposite requirements of confidentiality and verifiability. There is the further restriction of "fairness" that the intermediate election results must be kept secret. One of the ways to solve this problem is to use multiple layers of encryption such that the decryption key for each layer is available with a different authority. This obviously leaves open the question as to who is to be entrusted the encrypted votes.

A more effective way to implement fairness would be to avoid encryption keys altogether and divide each cast ballot into k or more pieces such that each authority is given one of the pieces [22]. This solves the problem of entrusting any single authority with all the votes and if any of the authorities (less than the threshold) try to cheat by deleting or modifying some of the cast ballots, then the votes may be recreated using the remaining partitions. Such a system implicitly provides a back-up for the votes.

6 Conclusions

We have introduced a new distributed data storage scheme for the sensor networks. In this scheme data is partitioned in such a way that each partition is implicitly secure and does not need to be encrypted. Reconstruction of the data requires access to a threshold number of sensors that store the data partition.

An additional variation to the scheme where the data partitions need to be brought together in a definite sequence may be devised. One way to accomplish it is by representing partition in the following manner: $p_1, p_1(p_2), p_2(p_3), ...$, where $p_i(p_j)$ represents the encryption of p_j by means of p_i. Such a scheme will increase the complexity of the brute-force decryption task for an adversary.

References

1. Perrig, A., Szewczyk, R., Wen, V., Culler, D., Tygar, J.: Spins: security protocols for sensor networks. In: Proceedings of ACM Mobile Computing and Networking (Mobicom 2001), pp. 189–199 (2001)
2. Eschenauer, L., Gligor, V.: A Key-management Scheme for Distributed Sensor Networks. In: The 9th ACM Conference on Computer and Communications Security, pp. 41–47 (2002)
3. Liu, D., Ning, P.: Establishing Pairwise Keys in Distributed Sensor Networks. In: The 10th ACM Conference on Computer and Communcations Security, pp. 52–61 (2003)
4. Zhu, S., Setia, S., Jajodia, S.: LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. In: 10th ACM conference on Computer and Communications Security, pp. 62–72 (2003)
5. Karlof, C., Wagner, D.: Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. In: 1st IEEE International Workshop Sensor Network Protocols and Applications, pp. 113–127 (2003)
6. Yin, C., Huang, S., Su, P., Gao, C.: Secure routing for large-scale wireless sensor networks. Communication Technology Proceedings 2, 1282–1286 (2003)
7. Demirkol, I., Alagoz, F., Delic, H., Ersoy, C.: Wireless Sensor Networks for Intrusion Detection: Packet Traffic Modeling. IEEE Communications Letters 10(1), 22–24 (2006)
8. Benson, Z., Freiling, F., Cholewinski, P.: Simple Evasive Data Storage in Sensor Networks. In: 17th IASTED International Conference on Parallel and Distributed Computing and Systems: First International Workshop on Distributed Algorithms and Applications for Wireless and Mobile Systems, pp. 779–784 (2005)
9. Kak, S.: On the method of puzzles for key distribution. International Journal of Computer and Information Science 14, 103–109 (1984)
10. Kak, S.: Exponentiation modulo a polynomial for data security. International Journal of Computer and Information Science 13, 337–346 (1983)
11. Shamir, A.: How to share a secret. Communication of ACM 22(11), 612–613 (1979)
12. Kak, S.: A cubic public-key transformation. Circuits, Systems and Signal Processing 26, 353–359 (2007)
13. Dickson, L.: Linear Groups with an Exposition of the Galois Field Theory. Dover Publications (1958)
14. Rosen, K.H.: Discrete Mathematics and its Applications. McGraw-Hill, New York (2007)