UNIVERSITY OF CINCINNATI

Date: 07/14/2006

I, Ananya Gupta,

hereby submit this work as part of the requirements for the degree of:

Master of Science

in:

Computer Engineering

It is entitled:

Decentralized Key Generation Scheme for Cellular-Based Heterogeneous Wireless Ad Hoc Networks

This work and its defense approved by:

Chair: Dr. Dharma P. Agrawal

Dr. Karen Tomko

Dr. Carla Purdy

Decentralized Key Generation Scheme for Cellular-Based Heterogeneous Wireless Ad Hoc Networks

A thesis submitted to the

Division of Research and Advanced Studies
Of the University of Cincinnati

In partial fulfillment of the
Requirements for the degree of

Master of Science

In the Department of

Electrical and Computer Engineering and Computer Science

Of the College of Engineering

July 2006

By

Ananya Gupta

B.E. (EE), Bombay University, Bombay, India 2001

Thesis Advisor and Committee Chair:

Dr. Dharma P. Agrawal

Abstract

A majority of group communication applications in cellular-based heterogeneous wireless setups entail secure data exchange. The problem can be effectively tackled if the underlying cellular infrastructure is used to provide an authentication backbone to the security associations. We propose a novel distributed ID based key exchange mechanism using shared polynomials in which the shares are generated by the communicating groups. Our idea employs a mechanism where the Base Stations (BSs) carry out an initial key generation by a polynomial in a distributed manner and then pass on the key material to the Mobile Stations (MSs). The multi-interface MSs can now securely communicate over interfaces other than cellular. The scheme incorporates symmetric polynomials, which are chosen by the BS acting as polynomial distributors. Simulations done to measure performance have shown encouraging results.

To My Parents, Brother and Beloved Fiancée

Acknowledgements

I wish to express my sincere gratitude to my advisor, Dr. Dharma P. Agrawal, for the leadership that he has provided me throughout my research. He has always strongly encouraged us to work in areas that personally interested us. I would like to thank Dr. Carla Purdy and Dr. Karen Tomko for being a part of my thesis committee.

In addition, I would like to thank all my lab mates, in particular, Anindo, Anurag, Lakshmi, and Torsha at the Center for Distributed and Mobile Computing (CDMC) for their constant support and helpful suggestions during the course of my thesis work.

My friends have made my life wonderful and memorable in Cincinnati. Special thanks to Akshay, Altaf, Andreas, Ashish, Hemant, Lynda, Mamoon, Mary, Matt and Naveen. I will cherish their friendship and all the good times we have shared.

I would also like to express special appreciation to the Singhals for their constant support, love and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

1 Introduction

1.1 Mobile Ad hoc Network

A primary characteristic of an ad hoc network is that basic communication channels can be established between collaborating mobile stations without the need of any fixed infrastructure. These mobile nodes, once within radio range, can communicate with each other, either directly or by using multi-hop message propagation techniques. This enables information

devices to move out from traditional, fixed, desktop scenarios to a more independent distribution. Many cutting-edge commercial, industrial and military applications [1] are being envisioned using these devises. However, this infrastructure-less organization, pervasive wireless medium and high mobility greatly increases the security risks of operating in such an environment.

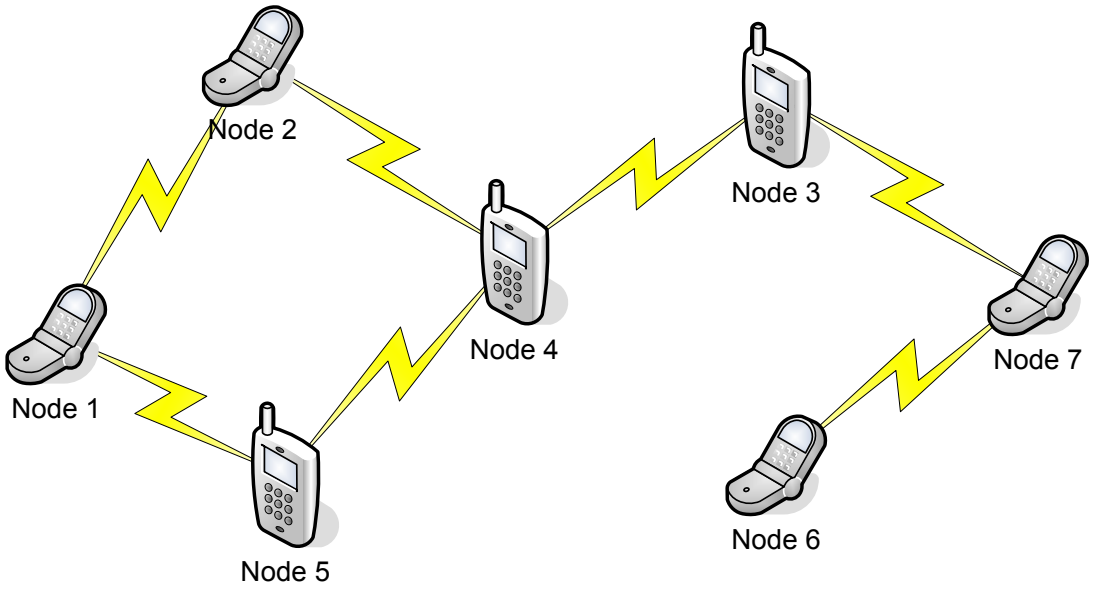

Figure 1 Mobile Ad hoc Network

1.2 Group Communication

As the number of Mobile Stations (MS) and their applications increase, the inherent desire to exchange data between them is also expanding. Group communication security in mobile ad hoc

and cellular networks has been of enormous interest in recent times. A group can be defined as a set of mobile nodes that wish to communicate with each other over a secured channel. The primary challenge facing a group key management infrastructure is that members may join or leave at any time. In either case, the key has to be appropriately refreshed, as it would otherwise enable an old member to be able to access the group data even when it is not a part of the group.

Thus, group key management requirements can be summarized [2] as: 1) Forward Secrecy: A node joining the network should not be able to compute the group keys that were in use prior to its join. 2) Backward Secrecy: A node leaving the group should not be able to compute new group keys that would be used in the network. 3) Key Independence: A node that is not a part of the group should not be able to derive any information about the group key from the knowledge of other group keys. 4) Group Key Secrecy: It is computationally infeasible for an adversary to derive the group key.

1.3 Encryption

There are two types of key-based encryption techniques that are widely in public use. One is asymmetric encryption using public-private keys and the other is symmetric encryption. Although the former provides a high level of security, for group communication, it increases the key management and communication overhead – a factor that becomes a crucial constraint in wireless ad hoc networks [1]. Alternatively, using a single key, shared between all the members for encrypting the group session is more efficient. However, one of the underlying problems addressed in this research, in using a symmetric key, is distributing the key and updating it between the group members securely and efficiently.

Hence, it follows that a primary issue in ad hoc scenarios is establishing trust within the group. A second issue is maintaining a secure channel between the trusted members. To initiate a trust between the ad hoc elements, we merge our key distribution scheme with elements of the cellular backbone. Ad hoc communication group-security involves problems concerning scalability of key generation and distribution. Additionally, the bandwidth overhead and energy use associated with group setup

must also be minimal due to constraints on both. For group security, it is imperative to maintain appropriate security associations between key generators, distributors and receivers ensuring a certain level of trust between them. This problem has been commonly approached in ways that logically segregate the key management/distribution entities and group member entities [3, 4].

1.4 Overview of Our Scheme

The group key exchange, management and distribution scheme introduced in this paper aims to facilitate secure ad hoc group communication using existing cellular infrastructure [5] as part of a pervasive distributed hierarchical trusted entity. In applying our scheme to ad hoc groups, we segregate wireless data exchange into two distinct domains – local ad hoc networks and larger domains using third party mediums such as a cellular network. Although a localized transfer supports higher transfer speeds, using a cellular network provides near-global coverage but at a much lower bandwidth. If a service such as General Packet Radio Service (GPRS) is used then it also involves a rolling cost attached to per unit bandwidth consumed in the data exchange process.

Many new Mobile Stations (MS), such as the HP iPAQ h6315 [6], are equipped to simultaneously interface with a variety of mediums – both in ad hoc (Bluetooth, Infrared, Wi-Fi) and infrastructure-based (cellular, access point) networks. With this development in view, the scheme presented in this paper aims to utilize the cellular network for key management. By using the cellular backbone for initial key setup and distribution, we are employing the inherent security association and trust between multi-interface MSs and the cellular network. Moreover, wide distributions of cellular Base Station (BS) networks ensure that allocating keying materials over its interface to such MS devices can take place almost anywhere. This is equally applicable if BSs are replaced by Access Points (APs), served by the same ISP, maintaining trusts with each mobile station. Each AP does the functionality of a BS as far as the group is concerned.

We adapt the cellular protocol and scheme so that a group of heterogeneous MSs (equipped with multiple interfaces, like IEEE 802.11) uses the cellular network for authentication and for obtaining the security keys in a distributed fashion through the BSs. Since we consider a localized domain, the devices are no longer limited to using the cellular network but may also use

higher bandwidth ad hoc Bluetooth/ Wi-Fi/ infrared links to initiate their encrypted group session.

The advantages of this approach are:

1. BS infrastructure allows parallel key distribution to participating ad hoc devices as needed say every 30 minutes
2. Cellular security association maintains inherent unique 3rd party upper-tier trust for each device
3. With requisite keying materials, an encrypted channel can be established between few selected MSs on any feasible ad hoc interface
4. Secured group is safe from any MSs that may have been authenticated by impersonating BSs

Although the key distribution scheme uses polynomial based encryption, which is vulnerable to threshold security issues, the member MSs are not particularly at risk due to the localized nature of the group communication. The encrypted wireless transmissions are assumed to be of short duration and over a narrow, localized, wireless domain. Moreover, a localized domain also permits a general re-key performed over the entire

group, say every 30 minutes, with comparatively little overhead due to its high-bandwidth nature.

1.5 Thesis Organization

Following the introduction, this thesis is organized as follows. In the next section, we will discuss security issues in information networks and establish the problem. In Chapter 3, we describe the main scheme, cellular authentication and also present methods to reduce the communication overhead. Chapter 4, provides a security and performance analysis. Some related work is presented in Chapter 5 while Chapter 6 presents the conclusions.

Chapter 2

2 Security

Whenever we talk about digital data networks as opposed to analogue links, one growing distinction is that typically, analogue communications are circuit-based setups, where the sender and receiver ‘own’ the link from point-to-point for the duration of information exchange. The current trends in Internet-based multimedia, voice and data applications implies that an increasing amount of shared and confidential information now travels over the very same open mediums which makes the

information vulnerable to a ‘digital hijack’ by a malicious entity. This section will attempt to provide an overview of information security and how it applies to data networks.

2.1 Why do we need security?

In today’s connected world, it is practically impossible to do business with customers and business partners without sharing personal or confidential information. Information exchanged may involve confidential finance data (e.g. uploading financial statements and check balances to company account holders), taking onus for customer charge-card information during an online order or intent to share confidential third-party information between team members at a public expo. A secured communication is successfully established between the engaging parties if the information shared is inaccessible to any entity external to the group [7, 8]. By inaccessible we imply that any information gleaned by the external party during the conversation is sufficiently encrypted so that it no longer conveys the original message, in whole, or in part [9].

Applying this precedent to the earlier mentioned scenarios immediately implies a widespread applicability of such a technique. However, a secure communication warrants certain prerequisite parameters to ensure its implementation and robustness [7, 8] – it must inherently comply to support the attributes of *Confidentiality*, *Integrity* and *Availability*.

2.2 Properties of a Secure Communication

2.2.1 Confidentiality

The basic premise of confidentiality is that intended message is 'visible' only to the original sender and expected receiver. Any eavesdropper should not be able to comprehend the intended message even if the entire conversation is overheard. The secure communication system in place should maintain the confidentiality irrespective of interception or interruption of the transmitted information.

2.2.2 Integrity

Integrity of the data transferred implies that the original information sent is unaltered and received "as-is" as the receiver.

As prerequisites, integrity maintenance must involve a form of authentication and non-repudiation.

- *Authentication*

  The scheme in place should be able to correctly identify and tag participating members and ensure that only intended members are involved in the secure communication. The scheme should take into consideration that the communicating members might encounter falsehood information and alteration of data.

- *Non-repudiation*

  The secure communication scheme should be in a position to irreversible account for the occurrence of an event in the medium. That is, any communication occurring between the engaging entities must be undeniably existent and verifiable. This is imperative to elevate trust levels between collaborating parties in the secure communication information exchange.

2.2.3 Availability

The secure channel should be available to all authorized members with minimal overhead and no compromise to confidentiality. There must be no access restrictions on the data transferred between intended parties while it exclusively prohibits all unauthorized entities. Owing to the nature of intent, logically, one might observe an inverse relationship between availability and confidentiality.

2.3 Network Attacks

Confidentiality of information arises from the presumption that its knowledge is directly proportional to the advantage of bargain. To execute present-day business and applications, at some point one will have to rely on third-party carriers for successful interactions. The basis of most security schemes is an assumption of trust between collaborating entities at some higher order of communication. It is safe to say that any network activity designed to counter these trust levels may be classified as a network attack.

These attacks are primarily of two types – passive and active with the later being more intrusive. As an analogy, a passive

attacker is like an eavesdropper who increases the gain of his microphone to simply obtain a copy of the information being shared with least detection and minimal invasiveness in the ongoing communication between the engaging parties. There is no fabrication or alteration of information towards the engaging parties by the passive attacker.

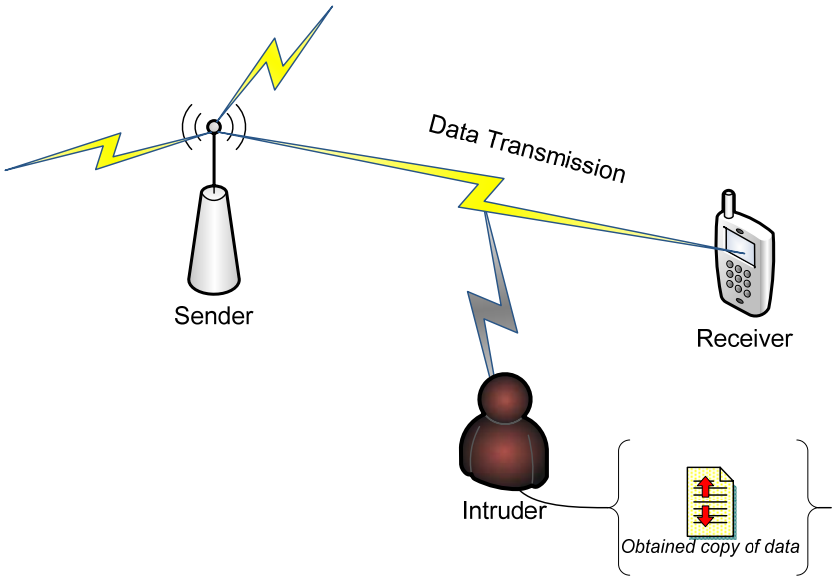

Figure 2 Passive Attacker

Based on the motives of the attacker it may scheme beyond just obtaining a copy of the information, such as further impede its receipt at the destination or inject malicious code into the original message altering its content or send an entirely fabricated message between the engaging parties. Malicious nodes acting along these grounds would be classified as active attackers.

Successful active attacks usually require the attacker to *masquerade* as one of the engaging parties in the communication involving control over certain network parameters. Alternatively, the attacker may be initially passive to obtain a copy of the information being exchanged and then choose to later *replay* this stored data over the network to his advantage to the point that his actions may cause network detrimental effects on the communicating medium. Depending upon the strategy, the attacker may also *alter* the original data before replaying it.

An entirely different kind of technique-based network attacks, which are more seriously damaging, are Denial of Service and Distributed Denial of Service attacks. The primary motive is the logical isolation of an authorized entity or set of entities from the network, using fundamental network-related or host application vulnerabilities.

2.3.1 Denial of Service (DoS)

The malicious node sends multiple, rapid, lengthy response-requests with the intention of overwhelming the receiving host. Once the receive queue or buffer of the host is filled with queries from the malicious node it will be unable to answer to any legitimate requests from other members communicating on the

network and instead it will start dropping their legitimate packets. Since they received no replies, other legitimate nodes will assume that the host either has dropped out of range, or is erratic.

2.3.2 Distributed Denial of Service (DDoS)

To counter the DoS, the host has now been programmed to drop response-requests originating from the same source and to stop servicing such a sender beyond a certain threshold. As a result, the malicious node now takes help of other legitimate members of the network to work against this newly enhanced host. Using other active attacking techniques, it takes control of a handful of member nodes and programs them to send requests to the intended host simultaneously at its command. In such a situation, the malicious node is the Master controller and the compromised nodes form the Slaves. At the signal of the Master node, the Slaves will send multiple request-responds to the host in parallel, disabling it from servicing any genuine queries.

2.4 Cryptography

Cryptography is the process that uses established algorithms to convert information into seemingly incomprehensible text – called

‘cipher-text’. As part of this algorithm, with the help of the requisite parameters, original information can once again be recovered. This algorithms used in this process are addressed as encryption algorithms. The strength of the cipher-texts depends upon the strength of the encrypting algorithm used.

The engaging entities willing to share confidential data over public mediums wish to do it in such a manner that only authorized parties can obtain the original message. Since the encrypted message or cipher-text will be transmitted over public mediums, beyond a certain capacity any enterprising individual who has access to the same medium can obtain a copy of the cipher-text. It will be the strength of the encrypting algorithm that will prevent him from deciphering the essence of the transmission. The science of studying encrypted texts to reverse engineer the encryption algorithms is called cryptanalysis. Unless the intruder successfully reverse engineers the cipher-text or ‘cracks’ it, the stolen data will be garbled junk.

The ancient Greek *scytale* ciphering is thought to be one of the earliest practical implementations of cryptography in use. Its principle is based on transposition of text in proportion to the

diameter of the cylinder around which a strip of paper is wound as illustrated in the diagram shown below.

Figure 3 Greek scytale ciphering

Although, this method is crude and easy to crack, modern day cryptography has come a long way since then.

2.5 Symmetric Cryptography

Also known as, conventional cryptography, this form of encryption utilizes the same key for both, encryption of the message and decryption of the cipher-t ext. This key is called the *symmetric-key* or *secret key* and the type of encryption is called symmetric-key encryption.

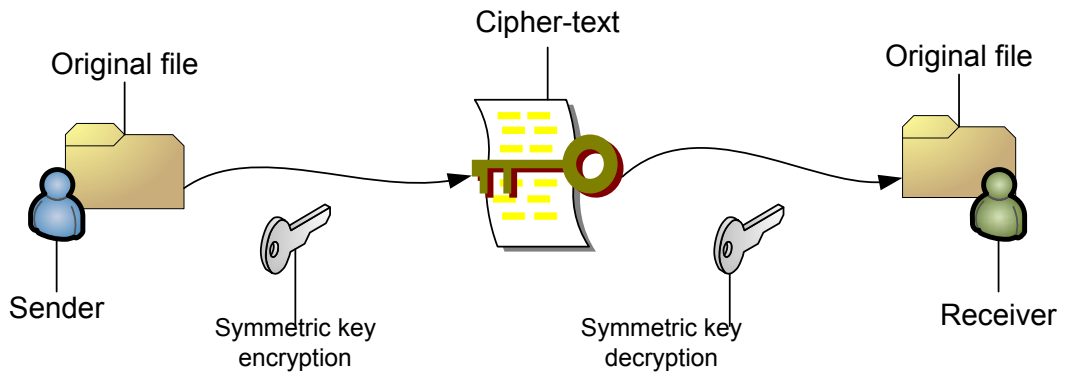

Figure 4 Encryption using symmetric key

The main drawback of symmetric key encryption is its inherent property of using the very same key employed during encryption at both ends; the sender and receiver for decryption. In the event that the engaging parties are geographically spaced, a third-party entity will have to be trusted for sharing the secret key. As a result, a compromise of this third-party entity will render any secure communication channel using symmetric key encryption completely useless.

2.6 Asymmetric Cryptography

This technique is different in that two different keys are used for encryption and decryption– one for each process. Also known as *public-key* encryption, the decryption key cannot be determined in any feasible duration from the encryption key. The encryption

key, known as the *public-key* is made known to everyone so that others can encrypt messages and send them to the holder of the decryption key known as the *private-key*. The private key is kept secret and is used by the recipient of the cipher-text to decode his received messages. At times, the private key may also be used to send out encrypted messages to be decrypted using the appropriate public key. This technique was first introduced by Whitfield Diffie and Martin Hellman, hence it is also known as the Diffie-Hellman encryption. The knowledge of public and private keys belonging to various entities is usually managed by a Certificate Authority, which also maintains a directory lookup feature.

When a user wants to send an encrypted message to another entity, it will first approach the Certificate Authority and lookup the public key for that entity on the directory service. It will then encrypt the message with the public key and transmit it over any communication medium available.

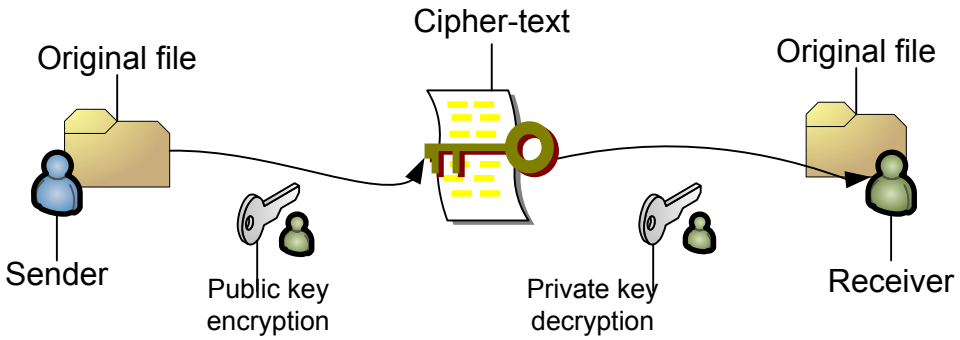

Figure 5 Encryption using public-key

At the receiver end, the entity will run the received cipher-text through its private key to obtain the original message.

2.7 Digital Signatures

As mentioned earlier, sometimes messages are sent out after encrypting them using the sender's private key. Such encryption offers non-repudiation, since only the holder of the private key, which is secret to that particular sender could have encrypted a message that can only be decrypted by the corresponding public key. This is akin to providing a unique signature for all to see. It also provides for authenticity of the sender as long as the private key is still secret only to the intended holder.

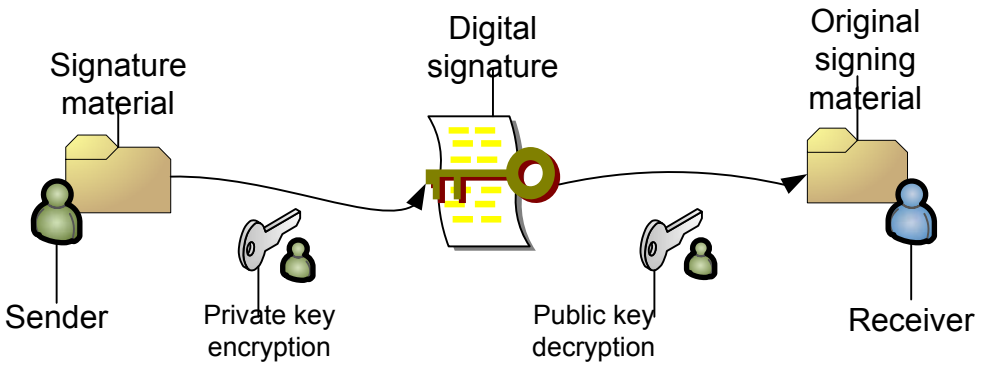

Figure 6 Digital signature

2.8 Digital Certificates

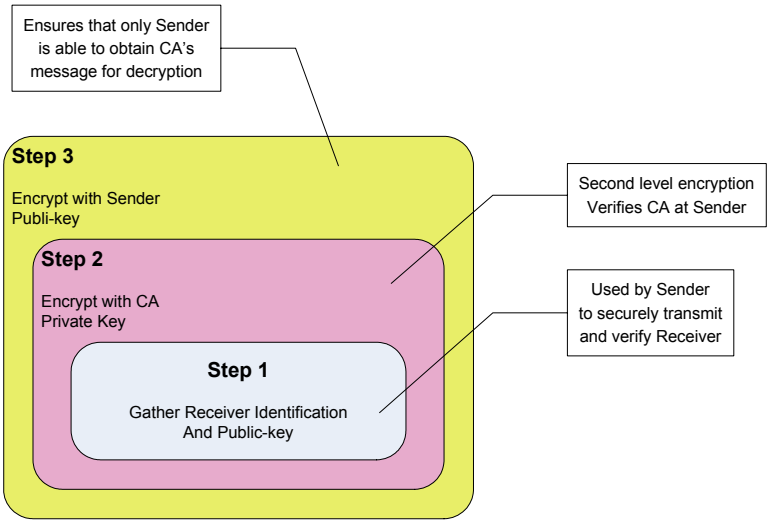

Figure 7 Using Digital Certificates

By taking digital signatures and public-key encryption one level further, using a third party Certification Authority (CA) one can

provide a more complete key management solution. This method allows indisputable identification of entities on a network. However, as a prerequisite, the entity in question must also be part of a pre-registered trusted relationship with the CA.

When a Sender wants to inquire about a potential recipient, the CA will provide identification information and public-key of the Recipient first encrypted with the CA private key and then re-encrypted with the Sender's public-key. The process is then reversed at the Sender to obtain the public-key of the potential Recipient.

2.9 Preliminaries and Problem Statement

We first define a few terms that would be used later in the text.

- A Node Group (NG) is a group of MSs with the same polynomial distributor and derives its keying material from this leader
- An Ad Hoc Node (AHN) is a MS that belongs to a NG
- Polynomial Distributor (PD) is a BS that acts as a polynomial supplier to a NG. A PD is a *founder*-PD if it was involved in the initial key generation process.

We now state the problem as follows: The need is to devise a hierarchical symmetric key generation scheme in which the PDs decide (in a manner explained in the next section) upon the key material and pass it on to their respective AHNs. Trust relationships are such that no AHN from another NG should be able to decipher any AHN conversation in which it does not participate. In the cellular heterogeneous scenario, considered in this paper, an NG corresponds to several MSs that wish to form a group. These MSs may or may not be under the same BS of same cellular service provider.

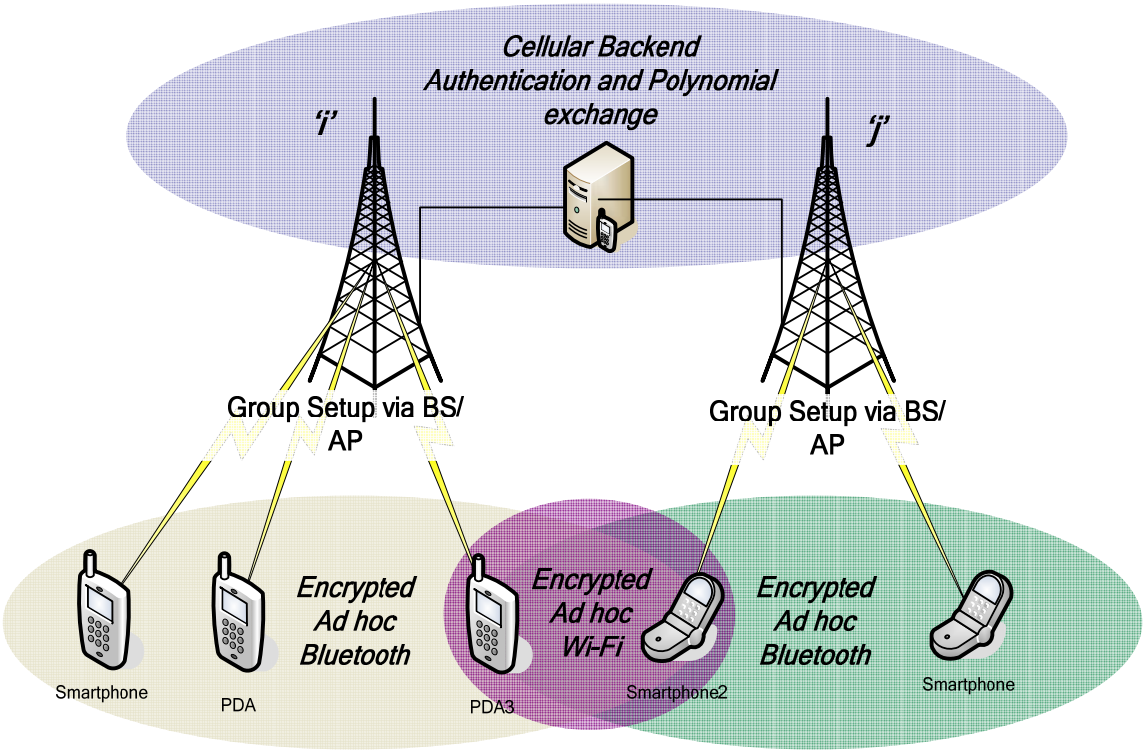

Figure 8 Decentralized Heterogeneous Ad Hoc Network

Consider the example in Figure 8. Owners of PDA3 and Smartphone2 wish to exchange confidential product drawings during a conference using ad hoc Wi-Fi. In such an environment, one problem, immediately apparent is that their transactions would be open to any uninvited prying MSs within the vicinity. Using our scheme, in such a scenario, the communicating MSs are provided with appropriate key generating polynomials through the BSs of their respective cellular service providers. As is explained in the next section, this setup allows the establishment of a basic pairwise secret relationship between the two (or more) MSs which collaborate amongst themselves. Communication using this secret pairwise setup can now independently take place securely over ad hoc Wi-Fi for securely distributing requisite group session-encryption keys.

Chapter 3

3 Proposed Scheme

3.1 Description

Our underlying key generation algorithm is based on the schemes suggested in [10] and [3]. The novelty of our scheme lies in making the scheme distributed. In essence, our scheme is implemented in three stages. The principal objective of the following scheme is to enable each MS to be able to securely communicate with any other

MS. This should be possible without any prior communication between the MSs. Once a secure pairwise channel is setup between any two MSs, group formation can be initiated without any further intervention from the BSs.

It should be noted that these keys are not the group session keys and are merely pairwise keys, which are then used to encrypt messages, including sending group session encryption keys, exchanged during the group setup process.

Phase 1:

The primary intention of this phase is to have a key distribution among the PDs. Once the polynomials are selected by a group of PDs, based on the criteria explained later, the PDs can provide the member MSs of corresponding NGs with supplementary pairwise keying material. This technique facilitates an independent creation of ad hoc groups by the collaborating MSs without any further BS intervention.

As stated earlier, the network consists of several BSs. Each of these BSs is a PD of a NG. Thus, each NG has a set of AHNs and a PD. We denote a participating group of MSs (AHNs) by NG_i and the

polynomial distributor of NG_i by PD_i, ($1 \leq i \leq n$), where n is the number of PDs. The j^{th} AHN belonging to a group i is denoted by AHN_{ji}. The size of a group NG_i is denoted by $|NG_i|$.

At the outset of phase 1, each PD_i chooses a function f_i in four variables w,x,y and z, such that:

$$f_i(w,x,y,z) = f_i(x,w,y,z) , \qquad (1)$$

and

$$f_i(w,x,y,z) = f_i(w,x,z,y). \qquad (2)$$

The variables w and x represent the MSs and y and z denote the variables associated with PDs. The maximum degree of this polynomial is t in each variable.

We note from (1) and (2) that the polynomials have to be symmetric in w,x and also in y,z. Thus, the coefficient of $w^m x^n$ should be the same as that of $x^m w^n$. In addition, the coefficient of $y^m z^n$ should be the same as that of $z^m y^n$.

The choice of the polynomials is dependent entirely on the PD. The robustness of the polynomial lies in the size of the coefficients and the degree of the polynomial.

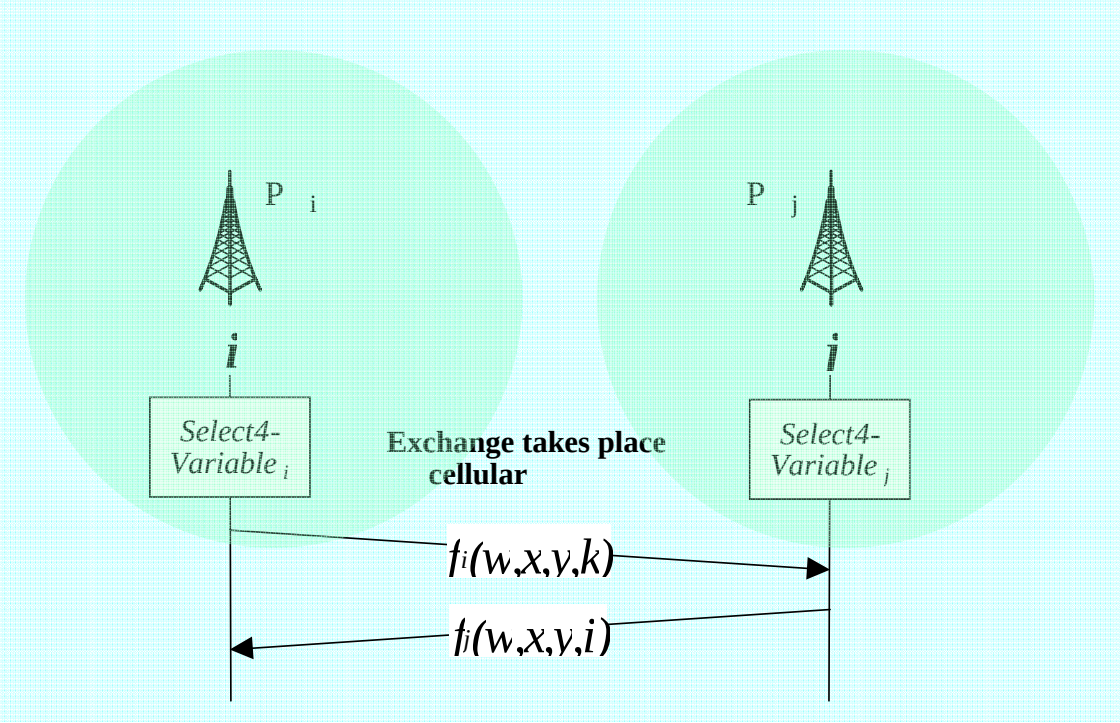

Figure 9 Phase 1 Setup: Polynomial exchange between PDs

After having chosen the function f_i, PD_i sends $f_i(w,x,y,k)$ to PD_k, which could be part of another cellular service provider, as shown in Figure 9. This communication takes place over secured, pre-authenticated, backend cellular channels. Each PD_i now obtains the polynomial P_i as follows:

$$P_i(w,x,y) = \sum_{k=1}^{n} f_k(w,x,y,i) \ldots \qquad (3)$$

Phase 2:

Once a PD_i obtains P_i, it evaluates it at the ID of each of its group member AHN_{ki} as:

$$S_{ki}(x,y) = P_i(ID(AHN_{ki}),x,y) . \qquad (4)$$

This quantity is now sent by PD_i to AHN_{ki} as represented in Figure 9.

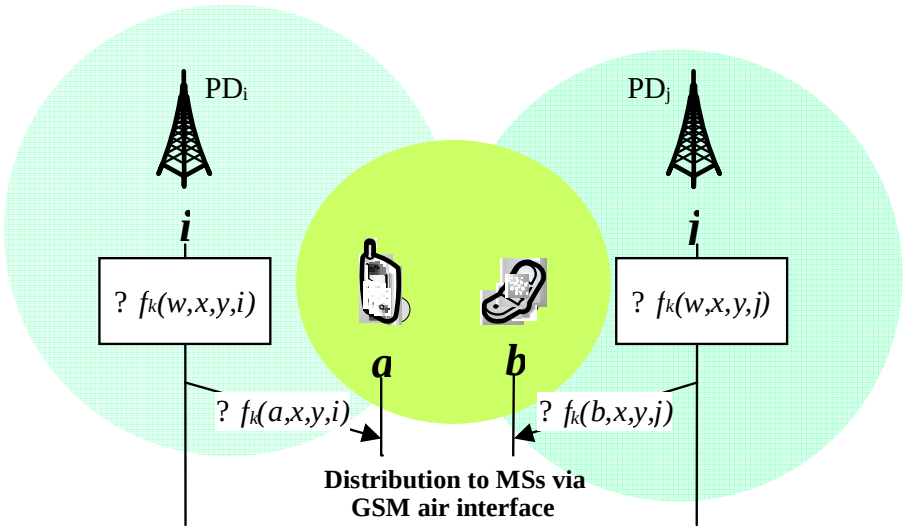

Figure 10 Phase 2 Setup: Distributing polynomials to MSs

The polynomial *S* is all that an AHN requires in order to compute the pairwise key with any other node in the ad hoc group. In order to calculate a pairwise symmetric key with any other node in the network, the node AHN_{ki} simply substitutes the ID of the PD of the other node for *y* and the ID of the other node for *x*.

Phase 3:

We demonstrate the key establishment process by considering the following example.

Let there be two PDs with IDs *i* and *j* and two MSs associated with these BSs with IDs *a* and *b*.

Thus, after Phase 2, node *a* receives the following as its key material:

$$S_{ai}(x, y) = \sum_{k=0}^{n} f_k(a, x, y, i) .$$

And node *b* receives: (4a)

$$S_{bj}(x, y) = \sum_{k=0}^{n} f_k(b, x, y, j) .$$

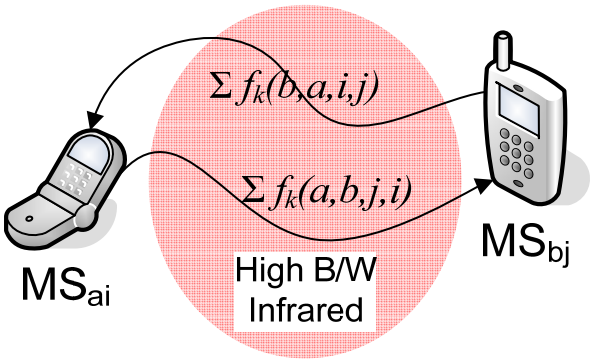

Figure 11 Phase 3 Pairwise direct message exchange between AHNs

As illustrated in Figure 11, if nodes *a* and *b* now wish to communicate with each other, node *b* calculates:

$$S_{bj}(a,i) = \sum_{k=0}^{n} f_k(b,a,i,j) .$$

And node *a* calculates: (4b)

$$S_{ai}(b,j) = \sum_{k=0}^{n} f_k(a,b,j,i) .$$

Using (1) and (2), we see that $f_k(a,b,i,j) = f_k(b,a,j,i)$, for every k, $1 \leq k \leq n$. Thus, $S_{bj}(a,i) = S_{aj}(b,j)$. Thus, nodes *a* and *b* would have the same symmetric key to communicate with each other. Further, this calculation is achieved without any message exchanges between *a* and *b*.

3.2 Efficiency Improvement at PDs

We now consider how to select the polynomials by the PDs so that optimal performance could be achieved. Let $\mathbf{A_i}$ be the coefficient matrix of $f_i(w,x,y,z)$ with dimensions (t^2, t^2), such that the $(\alpha, \beta)^{th}$ element of the coefficient matrix **A** denotes the coefficient of $w^k x^l y^m z^n$

Where,

$$l = \alpha / t, \qquad (5a)$$

$$n = \beta / t. \qquad (5b)$$

Also,

Let **R** be the matrix:

$$\mathbf{R} = [1, w, w^2, .., w^{t-1}, x, xw, .., xw^{t-1}, .., x^{t-1}, x^{t-1}w, .., x^{t-1}w^{t-1}].$$

And, let **Q** be the matrix:

$$\mathbf{Q} = [1, y, y^2, .., y^{t-1}, z, zy^2, .., zy^{t-1}, ..., z^{t-1}, z^{t-1}y, ..., z^{t-1}y^{t-1}].$$

In order to improve the efficiency, we choose *f* to be a sparse polynomial. Thus, instead of transmitting the entire coefficient matrix, from PDs/ to the MSs of a NG, now only an indexed array of the non-zero coefficients may be involved in the transmissions. Since this size is much smaller than the size of the entire coefficient matrix, the size of the messages can be drastically reduced.

As a downside, by employing sparse matrices, an attacker (in most cases, a compromised MS) can gain relative advantage by subjecting the system to a brute force attack. However, we observe that with a relatively large number of *PDs,* the final addition of the coefficient matrix would have a much-reduced sparseness. To verify this, let *s* denote the probability of the element (*α, β*) of the matrix $\mathbf{A_i}$ to be non-zero.

Thus, the probability *s'* of the element (*α, β*) of $\sum \mathbf{A_i}, (1 \le i \le n)$, being non-zero is *s'* = $(1-(1-s)^n)$.

Here, *n* is the number of *PDs* participating in Stage 1. This probability is plotted in Figure 12. As can be seen, with an increase in the value of *n*, this *s'* approaches 1. Thus, if the number of PDs in Stage 1 is large, the polynomials can be chosen

to be sparse. However, with a large number of PDs, the disadvantage is an increase in the number of messages. However, a large number of them need to go through the hardwired cellular backbone and should not be a major issue.

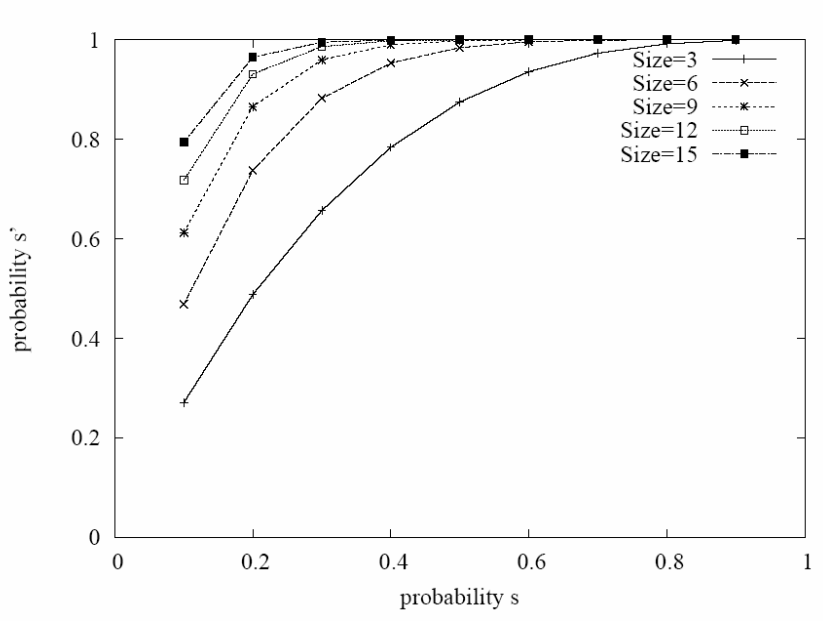

Figure 12 Plots for the probability s' of the element (α, β) being non-zero

We now present a technique to help choose the coefficients in a manner, which is both secure and efficient. In order to reduce the sparseness of the matrix $\mathbf{A}_i$, we do the following:

1. In the matrix **Q**, we randomly choose exactly one term out of the t terms $z^{i,}_{(0 \le i \le t-1)}$, corresponding to each term of y^j,

$(0 \le j \le t-1)$ with probability λ_1. We thus have a maximum of t terms chosen out of **Q**.

2. In the matrix **R**, we randomly choose exactly one term out of the t terms $w^i, (0 \le i \le t-1)$, corresponding to each term of x^j, $(0 \le j \le t-1)$ with probability λ_2. We thus have a maximum of t terms chosen out of **R.**
3. In the coefficient matrix $\mathbf{A_i}$, we allow a term $a_{\alpha,\beta}$ to be non-zero only if α corresponds to a row chosen in Prop. 1 and β corresponds to a column chosen in Prop. 2.

By following the above steps, the polynomials are evaluated for each individual AHN and the polynomial S is obtained as per (4) would have all terms as non-zero if λ_1 and λ_2 are 1. In future, we plan to analyze the case when λ_1 and λ_2 are not 1.

3.3 Group Establishment Protocol

As noted earlier, each MS can securely communicate with any other MS using the pairwise keys established in Section III.A.

The group leader is defined as the initiator of the localized heterogeneous ad hoc group. It generates the session encryption key, K_{S0}, for the defined duration.

Once the MS devices receive their individual keying materials, the group leader sends K_{S0} over a heterogeneous, high-bandwidth, local domain to the other group members. K_{S0} is then used for session encryption to forming a secure group channel between the members over any high-bandwidth local ad hoc interface.

In the event of a member leaving the secure group, the member leader generates a new session encryption key, K_{S1}, and sends it to each remaining member over the high-bandwidth localized domain. Alternatively, it can request complete polynomial re-keying from the PD for the entire group.

Likewise, a group join involves the creation of a new session encrypting key and distributing it to every group participating member including the newer joining entities.

Additionally, it is important to consider the case of a potential member authenticated by an impersonating base station, *BS'*. Since such a *BS'* is outside the cellular infrastructure, it is unable to provide the necessary keying materials that would enable a MS to directly contact another group member.

Chapter 4

4 Protocol Analysis

4.1 Security Analysis

As the security of the group formation mechanism depends solely on the pairwise keys used to encrypt the group formation messages, we focus our analysis on the security of the pairwise keys.

The security analysis of the pairwise keys depends upon the inherent security of the key distribution scheme. In any hierarchical threshold system, the number of colluding nodes differs at each level. This is primarily because of the unevenness in the distribution of the tiers. In what follows, we look into the number of nodes required at each level to carry out a successful collusion attack.

We define *compromise* of the system as the situation where a node (AHN or PD) becomes aware of a polynomial (*f, P* or *S*) which it should not know. Table 1 illustrates the various combinations of nodes required to launch a successful attack.

Entity Type	Compromise conditions	Compromised polynomial	Same group
m *PDs*	$m \geq t$	$f(w,x,y,z)$	–
m *AHNs*	$m \geq t^2$	$f(w,x,y,z)$	–
m *AHNs*	$m \geq t$	$S(x,y)$ $P(w,x,y)$	Yes
a *PDs and* b *AHNs* ($a \leq t$, $b \leq t^2$)	$at + b \geq t^2$	$f(w,x,y,z)$	No

Table 1 Security analysis of our scheme - The 'Entity Type' column represents colluding entries

Lemma 1: Our scheme is secure to the collusion of a maximum of *t-1* nodes of any kind.

Proof: We first observe that each PD knows t^3 equations while each AHN knows t^2 equations for the coefficients matrix $\sum \mathbf{A_i}, (0 \le i \le t-1)$. Further, if AHNs and PDs collude, the AHNs belonging to the same NG as a PD would contribute nothing to the attack.

We now consider collusion between PDs. Since each PD is aware of t^3 equations, a PD would require *t-1* other PDs to collude so as to get all the coefficients of the polynomial $f(w,x,y,z)$,.

Next, we consider an attack by an AHN node. Since the polynomial $S(x,y)$ provided to an AHN, has been evaluated at two points, collusion between AHNs would require t^2 nodes to find out the polynomial $f(w,x,y,z)$. If the AHNs of the same group wish to attack another node of the same group or compromise their PD, collusion between atleast *t* nodes would be required.

Finally, we come to the case where *a* PDs and *b* AHNs collude. Since a PD contributes t^3 equations and an AHN contributes t^2 equations to solve for the coefficient matrix $\sum \mathbf{A_i}$, this combination can compromise the polynomial $f(w,x,y,z)$ if $at+b > t^2$. Solving for the inequality shows that $a+b$ is always greater than or equal to *t.*

We therefore infer that under no conditions can collusion help if the number of colluding members is less than t.

Hence Proved. □

Next, we consider a brute force attack. We see that there are t^2 unknowns in the polynomial $S(x,y)$. Thus, to carry out a brute force attack, all these values have to be guessed. Assuming a field of size κ for the polynomial coefficients, the attack complexity becomes κ^{t^2}. For $\kappa = 2^8$, this value is beyond the reach of all modern computers, even for a small value of t.

If sparse matrices are chosen as mentioned in the last section, the attacker would have no relative advantage, as all possibilities would still be equally probable.

4.2 Performance Analysis

The purpose of this analysis is to evaluate the scheme for the message and latency overhead. To keep the analysis generic and to incorporate scenarios involving WLAN APs as well as BSs, we

have simulated the network over an ad hoc network running a routing protocol.

We carried out our simulation in ns-2 [6]. The number of nodes varied from 30 to 70 and the number of groups varied from 3 to 15. The simulation area was taken to be 1000m X 1000m with a communication radius of each node as 200 meters. We fixed the speed of the nodes at 10 m/s. Finally, we used Dynamic Source Routing (DSR) as the underlying routing protocol. Although in a practical scenario, members belonging to the same NG would be located next to each other, we have allowed the AHNs to be distributed over the entire network area irrespective of the NG they belong to. The reason for doing this is to observe the worst-case performance of our scheme.

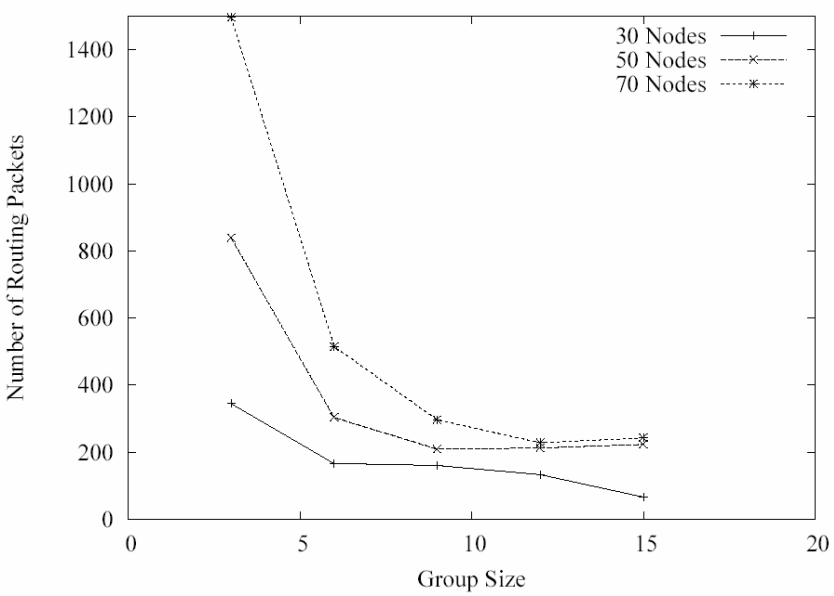

Figure 13 Routing overhead in key exchange

We first observed the effects of the size of the NGs over the routing overhead. Theoretically, with small NG sizes, the number of routing messages should be high, as more number of PDs would need to communicate with each other. Figure 13 plots the effects of the NG size to the overall network routing messages during the key exchange process. We notice that with an NG size of 3, the number of routing messages is very high. However, as the sizes increase, the routing overhead falls drastically and finally stabilizes. We attribute the reason for this to the fact that for smaller groups, the number of groups would be higher and thus, the initial $O(n^2)$ communication between the PDs would incur a very large routing overhead.

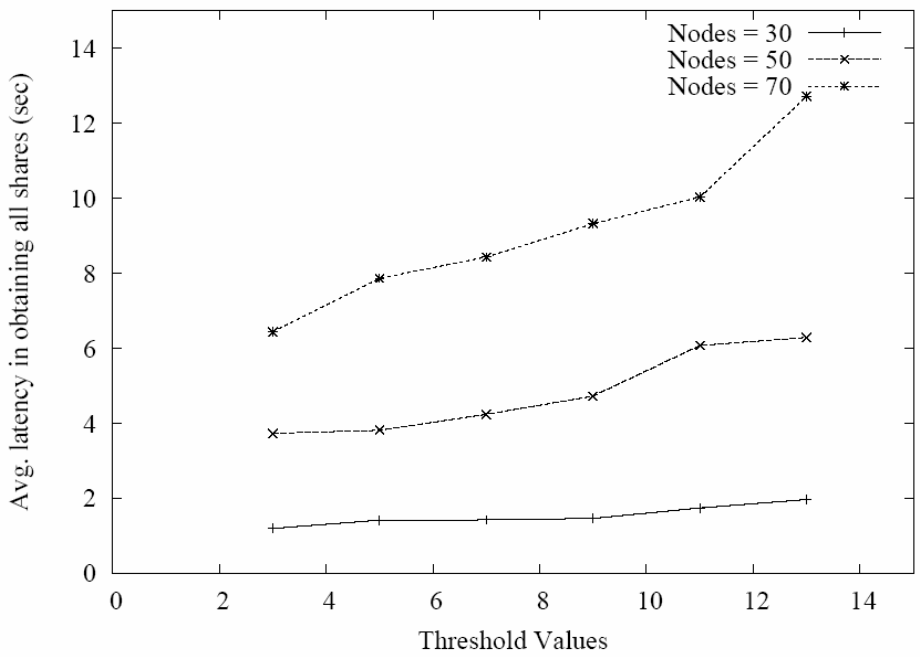

Figure 14 Average latency during the initial key exchange process

Next, we consider the effects of the threshold value over the latency in obtaining the key shares. A higher threshold value would lead to a higher number of coefficients in the polynomial and thus the message sizes would increase. As shown in Figure 14, an increase in the threshold values (t) leads to an increase in the latency values. However, the rise in the latency values is significant for a larger network. This is because in a smaller network, the messages do not need to a travel large number of hops and thus the delays are not significant.

5 Related Work

Key exchange using polynomials has always been an active area of interest. Several schemes have been suggested to exploit the symmetry in various kinds of polynomials [3, 10, 11].

Blom [10] discusses a symmetric key distribution scheme based on bivariate polynomials. This scheme is not distributed and relies on a central server to provide for the coefficients of a polynomial $f(x,y)$ evaluated at $x=i$, where i is the identity of the node. The function f

satisfies the property $f(x,y) = f(y,x)$. Thus, whenever two nodes wish to communicate, they evaluate their individual polynomials at the corresponding IDs of each other to get the symmetric key. For example, node i having $f(i,y)$ and node j having $f(j,y)$ would calculate $f(i,j)$ and $f(j,i)$ respectively. Due to the symmetry of the polynomial $f(x,y)$, these two quantities would be equal and would serve as the symmetric key between these two nodes. This idea has also been used by Liu et al. in [12] to devise a security scheme for sensor networks.

Blundo et al. [3] have analyzed symmetric polynomial schemes and suggested a hierarchical mechanism. However, in both [10] and [3] a central key distribution server has been assumed to be present for all key generation processes.

Other distributed key generation systems have been proposed in literature ([13] for example). Again, these schemes are not entirely distributed, as they require the presence of a centralized server to initially distribute the key generation materials. Kong et al. [13], introduce a scheme in which a distributed certificate is generated for each node in such a manner that a few nodes initialized with the key generation material send shares of the certificate to the requesting node. If the number of obtained shares is more than a

particular threshold, the requesting node is able to calculate the certificate.

Finally, Deng et al. [14] presents a distributed ID-based key generation scheme. A central server distributes the shares for a master key, based on which the individual shares for a node can be calculated in a distributed manner.

Chapter 6

6 Conclusions

In this work, we have proposed a novel method for a fully distributed key management and distribution technique in ad hoc networks for group communication, assuming MSs to be heterogeneous by having a second radio for access to cellular BSs. The implicit trust for all communicating MSs is derived from the trust that a MS enjoys with the backbone cellular network. In our scheme, we distribute key material to each node over this network

in such a manner that any two MSs can communicate securely with each other.

Once this trust has been established, a group formation can be initiated, by any MS with other MSs, without further intervention from the BSs. Further, these MSs can then communicate with each other over any interface and need not require the cellular interface for communication.

In a practical scenario, MSs having Bluetooth interfaces can form secure localized groups and communicate with each other, while enjoying the same trust as is provided in the underlying cellular network.

Security analysis of our scheme shows that it is robust to the collusion of a fixed number of nodes. However, by keeping the threshold values of the chosen polynomials high, collusion attack probabilities can be drastically reduced.

Bibliography

[1] Akyildiz, I.F., Su, W., and Sankarasubramaniam, Y., "A Survey on Sensor Networks," *IEEE Communications Magazine,* 2002,

[2] Kim, Y., Perrig, A., and Tsudik, G., "Simple and fault-tolerant key agreement for dynamic collaborative groups," *Proceedings of the 7th ACM Conference in Computer and Communication Security,* 2000, pp. 235-241.

[3] Blundo, C., De Santis, A., and Herzberg, A., "Perfectly-secure key distribution for dynamic conferences," *Advances in Cryptology – CRYPTO '92, Lecture Notes in Computer Science,* 1993,

[4] Mitra, S., "Iolus: A Framework For Scalable Secure Multicasting," *Proceedings of the ACM SIGCOMM 1997,* Cannes, France, 1997, pp. 277-288.

[5] Biot, O., "Global System for MObile Communication,"

[6] Hewlett-Packard, "HP iPAQ h6315 Specifications,"

[7] Agrawal, D.P., and Zeng, Q.A., "Introduction to Wireless and Mobile Systems," Brooks/Cole, 2002,

[8] Kurose, J., and Ross, K., "Computer Networking - A Top Down Approach," Addison Wesley, 2003,

[9] Shamir, A., "How to Share a Secret," *Communications of the ACM,* Vol. 22, 1979, pp. 612-613.

[10] Blom, R., "An Optimal Class of Symmetric Key Generation Systems," *Advances in Cryptology: Proceedings of Eurocrypt 84, Lecture Notes in Computer Science,* Vol. 209, 1984,

[11] Fall, K., and Varadhanm, E., "The ns Manual (Formerly ns Notes and Documentation)," 2000,

[12] Liu, D., and Ning, P., "Establishing Pairwise Keys in Distributed Sensor Networks," *10th ACM Conference on Computer and Communications Security (CCS '03),* 2003,

[13] Kong, J., Zerfos, P., and Luo, H., "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," *Proceedings of the IEEE 9th International Conference on Network Protocols (ICNP'01),* 2001,

[14] Deng, H., Mukherjee, A., and Agrawal, D.P., "Threshold and Identity-based Key Management and Authentication for Wireless Ad Hoc Networks," *IEEE International Conferences on Information Technology (ITCC'04),* 2004,