ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

TIK Institut für Technische Informatik und Kommunikationsnetze

Divya Sharma

Security of Field Based Routing

Student Thesis SA-2008-08
February 2008 to July 2008

Advisor: Mario Strasser
Supervisor: Prof. Dr. Bernhard Plattner

Abstract

Existing unicast routing protocols like, for instance, Ad hoc On demand Distance Vector routing (AODV) are not well suited for wireless mesh networks where most traffic flows between a large number of mobile nodes and a few access points that provide Internet connectivity and in other such application scenarios where packets are routed via anycast routing. Therefore, two (anycast) routing schemes based on Field Based Routing (FBR) have recently been proposed. In field based routing, each node is assigned a potential or field value and all nodes keep track of the potential values of their neighboring nodes. Access points and base-stations set their potential to infinity and nodes which are closer to an access point have a higher field value. Using these fields, packet forwarding is fairly simple: packets are forwarded along the nodes with the highest values until they eventually reach any access point.

Owing to their need for cooperative network operation, FBR is, like most ad hoc routing protocols, susceptible to routing attacks as well as malicious and selfish node behavior. Typical examples of attacks on ad hoc network routing protocols are: inserting spoofed, altered, or eavesdropped routing information; selective forwarding (gray holes); sinkhole/black hole attacks; wormholes; Denial of Service (DoS) and flooding attacks.

The main tasks of this project are to identify general as well as specific attacks on the FBR protocols and to propose counter-measures in order to avoid the attacks or alleviate their impact. In short, to develop a secure field based routing scheme.

We propose a novel taxonomy for classifying the attacker classes and the subsequent attacks which can be mounted on the system. We also propose two approaches for verification of the potential values and integrate these with the other available security mechanisms in order to suggest five different approaches which can be used to secure the field theoretic approach.

Acknowledgment

The author is highly grateful to her advisor, Mario Strasser, for his guidance and endless patience during the course of the project. The author also wishes to express her sincere gratitude to Professor Dr. Bernhard Plattner for his invaluable insights during the course of the project and for giving her a chance to work in a highly motivating and prestigious research environment.

Thank you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Ad hoc networks consist of nodes operating in a network without any fixed infrastructure like base stations or a centralized authority. Mobile nodes that are within each other's radio range communicate directly via wireless links, while those that are far apart rely on other nodes to relay messages as routers, that is, nodes in an ad hoc network cooperate with each other. Node mobility in an ad hoc network causes frequent changes of the network topology [55]. These networks differ from the wired networks where only specific nodes perform the task of routing or from managed wireless networks where a centralized authority manages the communication between the nodes.

Wireless ad hoc networks have acquired importance recently because of their potential to be used in various applications. These networks can be quickly deployed and can operate in areas where there is no infrastructure or a central authority to manage communication. These networks can prove to be highly useful for emergency situations like natural disasters or military conflicts. Hence, a secure and dynamic routing protocol is a prerequisite to ensure the proper working of these networks.

The two main networks we study under multihop wireless networks are:

- Mobile Ad hoc Networks (MANETs)
- Wireless Mesh Networks (WMNs)

A MANET (Mobile Ad hoc Network) is a collection of mobile nodes in which nodes cooperate in order to successfully transmit packets in the absence of a centralized authority or existing infrastructure [11].

A wireless mesh network is a communication network in which some nodes are configured as access points or mesh routers and these nodes communicate with the stations in the network. According to [56], *'Wireless mesh networks (WMNs) are dynamically self-organized and self-configured, with the nodes in the network automatically establishing an ad hoc network and maintaining the mesh connectivity.'* WMNs are comprised of two types of nodes: mesh routers and mesh clients. Mesh networks are created with a combination of stations and access points connecting each other via the wireless medium with an aim of creating a wireless distribution system.

Existing unicast routing protocols like, for instance, AODV (Ad hoc On demand Distance Vector routing) are not well suited for wireless multihop networks where most traffic flows are between a large number of mobile nodes and a few access points that provide Internet connectivity [4] and in other such application scenarios where packets are routed via anycast routing. Therefore, two (anycast) routing schemes based on FBR (Field Based Routing) have recently been proposed. In field based routing, each node is assigned a potential or field value and all nodes keep track of the potential values of their neighboring nodes. Access points and base stations

set their potential to infinity and nodes which are closer to an access point have a higher field value. Using these fields, packet forwarding is fairly simple: packets are forwarded along the nodes with the highest values until they eventually reach any access point [6].

Owing to their need for cooperative network operation, FBR is, like most ad hoc routing protocols, susceptible to routing attacks as well as malicious and selfish node behavior. Typical examples of attacks on ad hoc network routing protocols are: inserting spoofed, altered, or eavesdropped routing information; selective forwarding (gray hole); sinkhole/black hole attacks; wormholes; Denial of Service (DoS) and flooding attacks [53].

1.1 Motivation

The increasing demand for access to various resources even in areas where a centralized infrastructure is absent or in case of mobility, has led to widespread increase in the deployment of wireless ad hoc networks. As discussed in [6], 'There has been a tremendous increase in the recent years in the number of wireless ad hoc and autonomous networks that have come up around the movement of people in buildings, public vehicles, conferences or even in cities. Even though various fixed wireless infrastructures have served the purpose, there is still a need for ad hoc networks which do not rely on the centralized servers for their operation.' There are numerous other factors that serve as motivation for ad hoc networks, as discussed in [6]:

- Autonomous systems
- Networks not reliable on fixed base station
- High cost incurred for fixed wireless infrastructure
- Utility as the means of communication during disaster
- Reduction in the energy consumption
- Usage in poor/discriminated areas

1.1.1 Applications

There are a number of application areas in which the field based approach can be used. For instance, it can be used to provide service access [1]. Using this approach, specific services can be provided to the users. Field based approach can also be used for routing in sensor networks [2] as well as vehicular networks. Its applicability has also been studied in mesh networks [3], multihop networks that provide Internet connectivity to mobile users.

1.1.2 Motivation for secure services in multihop wireless networks

While the routing aspects of multihop wireless networks (e.g. MANETs, mesh networks etc.) are well understood, the research activities about the security are still at their beginning stages. In addition to the classical threats posed to the routing protocols, routing in multihop wireless networks is more susceptible to attacks. The provision of security services in these networks faces more challenges due to the nature of ad hoc networks [6]:

- Insecure wireless links
- Energy constraints and limited resources available in most cases
- Fixed centralized infrastructure might not be available
- All nodes in an ad hoc network are required to cooperate with each other and often no prior security association can be assumed
- Nodes may be mobile

Thus, the routing protocols in multihop wireless networks are vulnerable to a number of attacks [53] as will be discussed later in Section 5.2.

The field theoretic approach is susceptible to attacks as well, as will be discussed in Section 5.4.1. Indication of a wrong potential value in either of the application scenarios can disrupt the working of the routing protocol. For instance, in a wireless mesh network, if a node indicates a wrong potential value, it can cause all the packets intended for a gateway node to be routed towards it. In a sensor network, an adversarial node can cause excessive resource consumption of the already resource constrained devices by manipulating the field set up process. If the field theoretic approach is used for service discovery, the adversarial node can cause the service queries and replies to be routed to another destination than the sink it is intended for, by manipulating the field set up. The adversary can adversely affect the working of a protocol in these scenarios, and in the process it may eliminate the benefits the field theoretic approach offers. Hence the need to secure the routing protocol.

1.2 The Goal and the Tasks

The goal is to achieve a secure field theoretic approach which can be used in various application scenarios as mentioned above. In order to achieve this, there are a number of tasks required to be undertaken, which are:

- Become acquainted with the existing FBR protocols.
- Study well-known (general) attacks on ad hoc routing protocols and analyze their applicability to FBR.
- Find new FBR-specific attacks.
- Identify existing and devise new countermeasures/solutions (i.e. develop a secure field based routing protocol).
- Show their correctness and performance using (formal) security proofs.

1.3 Contributions

The main contributions of this work are as follows:

- The main contribution of our work is the identification of various attacks which can be mounted on the field theoretic approach in Section 5.4.1. We review the attacks on other ad hoc routing protocols and study their applicability to the field theoretic approach.
- We study the basic concept of FBR and identify three different phases from the security point of view, which are, *Establishment of potential field, Gradient determination and Packet forwarding.* We propose a system model in Chapter 4 in order to study the working of the protocol and the attacks that can be mounted on it.
- Based on the various attacks, we propose a novel taxonomy for classifying the attacker classes. These are, as discussed in Section 5.4.1, *Single adversary/multiple non colluding non neighboring adversaries, Multiple neighboring adversarial nodes/chain of adversarial nodes and Multiple colluding nodes.*
- We review various security mechanisms that have been proposed in literature in order to secure the protocols against specific attacks. These are discussed in Section 6.2 and a brief overview is given in Table 6.1.
- We propose two novel techniques for potential verification: local verification of potentials by showing the messages used in order to compute the potential as a proof; and the cluster approach. These are discussed in Section 6.2.2.
- Based on the available security mechanisms and the potential verification techniques, we propose five different approaches in Section 6.3, which can be used to secure the field theoretic approach. We analyse the approaches in Section 7.2 and give the performance evaluation in terms of control packet overhead in Chapter 8.

1.4 Overview

The rest of the report is organized as follows:

- Chapter 2 gives an overview of the related work done in the field of securing ad hoc networks. We discuss certain concepts which are fundamental to the understanding of the field theoretic approach. Firstly, we discuss the concept of routing and give a brief overview of the routing approaches in mobile ad hoc networks. Next, we give a brief review of the work which has been done in the field of securing routing protocols for ad hoc networks and Wireless Sensor Networks (WSNs).
- Chapter 3 gives a brief overview of field theoretic approach. We describe the field theoretic approach and identify three distinct aspects which are, namely, establishment of potential field, gradient determination and packet forwarding.
- Chapter 4 describes the system model which will be used to describe the protocol. We list the basic assumptions we make and give the protocol description, which shall be used for the following chapters. We also discuss the fields present in a typical routing message according to the protocol description.
- Chapter 5 describes the attacker model. We list the major security objectives we are working towards. We briefly describe the different attacks which can be mounted on the ad hoc routing protocols in general. There are mainly two classes of attackers, that is, the broad classification into Outsider attacker or Insider attacker, in accordance with the information available to the attacker and the types of attacks she can mount on the system. With reference to the field theoretic approach, we further classify the insider attacker into three main attacker classes: *Single adversary/multiple non colluding non neighboring adversaries, Multiple neighboring adversarial nodes/chain of adversarial nodes and Multiple colluding nodes.* We describe the different attacks which can be mounted by these attackers on the field theoretic approach and describe the impact of these attacks on the protocol.
- The secure protocol is proposed in Chapter 6. Firstly, we present the measures to secure the protocol against the Outsider attacker. Next, for the purpose of securing the protocol against the Insider attacker, we discuss the various countermeasures in general, which can be used to safeguard the field theoretic approach against specific attacks. We propose two main approaches for the purpose of verification of potential values: local potential verification by showing messages used in the potential calculation as proof; and the cluster approach. By combining the security mechanisms, we propose five different approaches which can be implemented in order to secure the field set up in FBR. Some approaches which can be used specifically for the sensor networks are discussed in the concluding section.
- The security analysis for both the models has been presented in Chapter 7. We analyse the countermeasures proposed for securing the protocol against the outsider attacker. A provable security analysis for the same is also presented. The analysis for the five secure approaches is also given.
- The results obtained are presented in Chapter 8. The performance evaluation of the secure approaches with respect to control packet overhead and the message size is given.
- The report concludes with a brief summary in Chapter 9 which also lists some directions for future work.

Chapter 2

Related Work

In this chapter, we give a brief overview of some concepts fundamental to understanding the nature of the field theoretic approach. Firstly, we describe the concept of routing, followed by a description of the different routing approaches which have been adopted in various networks. Next, we describe the main approaches which have been adopted for the purpose of routing in ad hoc networks. The various application scenarios where the use of field theoretic approach has been studied, have been reviewed. We also discuss the work accomplished in the field of securing ad hoc routing protocols.

2.1 Fundamentals

2.1.1 Routing

The process of determining systematically how to forward messages towards the destination node based on its address is called routing [9]. Routing is a fundamental component and is essential for a network to function, hence the target for various attacks.

There are various issues a routing protocol must address, as have been discussed in [9], for instance, ensuring the absence of loops, detecting node failure, minimizing the overhead network traffic and so forth. As discussed in [9], *Routing, as different from forwarding is the process by which forwarding tables are built to allow the correct output for a packet to be determined.* Routing Table needs to be optimized for the purpose of calculating changes in topology.

There are various characteristics of ad hoc networks, as discussed in [8] that make the task of routing challenging. Some of these are:

- Resource poor devices
- Limited bandwidths
- High error rates
- Continually changing topology
- Vulnerability to external attacks

Therefore, as the authors in [8] have discussed, the typical design goals for ad hoc network protocols include:

- Minimal control overhead
- Minimal processing overhead
- Multihop routing capability

- Dynamic topology maintenance
- Loop prevention

In the next section we describe some of the approaches which have been used for routing in mobile ad hoc networks.

2.2 Routing approaches in mobile ad hoc networks

There are basically two types of routing approaches in mobile ad hoc networks: proactive and reactive [8]:

2.2.1 Proactive approach

- Each node in the network maintains a route to every other node in the network at all times. Due to the instant availability of routes, there is less latency involved. However the storage of routes imposes a great amount of overhead on the nodes.
- Two main classes of Routing Protocols:
  - Distance vector
  - Link state
- The basic idea behind link state routing is that each node is aware of its immediate neighbors, and if this information is disseminated to every node in the network, then every node shall have enough information to build the complete map of the network [9]. Thus the shortest path to any node in the network can be calculated on the basis of the connectivity map.
- This contrasts with distance vector routing protocols, in which every node constructs a one dimensional array containing the 'distances' (costs) to all other nodes and distributes that vector to its immediate neighbors [9]. By having each node share its routing table with its neighbors, each node builds updated routing tables which decide the next hop for any packet to be routed.
- In a link state protocol, the only information passed between the nodes is information used to construct the connectivity maps. Compared to link state protocols, distance vector routing protocols have less computational complexity and message overhead.
- A number of different protocols have been proposed which use this approach, for instance, OLSR [12].

2.2.2 Reactive approach

- In this approach the route discovery is done via wide flooding of a request and routes are discovered only when they are really needed. This approach has an advantage over the proactive approach that the signaling overhead is greatly reduced relative to the proactive approach. The disadvantage is the introduction of route acquisition latency [8].
- Different protocols which have been proposed under this approach include AODV (Ad hoc On Demand Distance Vector Routing) [11] and DSR (Dynamic Source Routing) [13].
- Many other protocols using the *Geographical approach* or the *Hybrid approach* have also been proposed [8].
- Several secured ad hoc network routing protocols have been proposed, which shall be discussed in Section 2.4.1. Some of them have weaknesses that are exploitable by attacks [53].

2.3 Field Based Routing

The field theoretic approach has been studied in a number of different application scenarios. For instance, the approach has been used for anycast routing [6] where the trade off between proximity and density, which is inherent in field based routing, is exploited.

The concept has been studied in the context of density based routing and the authors in [4] have proved that the best performance lies in a trade off between proximity and density, hence the applicability of field theoretic approach to this context.

The concept of field based routing has also been studied for service discovery in mobile ad hoc networks [1]. The service providers or the destination nodes are modeled as positive charges and the potential field is established in the network. Service queries are assumed as negative charges which are routed along the steepest gradient, according to the established field.

Due to its applicability in the area of anycast routing, field theoretic approach has also been studied in the context of sensor networks [2]. In sensor networks, field based routing has been used to propose a routing scheme that ensures that the monitoring network remains connected and hence the live sensor nodes deliver data for a longer duration [2].

2.4 Secure ad hoc routing

2.4.1 Secure ad hoc routing protocols

A lot of research has been done in the field of securing routing protocols for ad hoc networks. There have been a substantial number of surveys regarding the attacks which can be mounted on ad hoc systems. Hu and Perrig have comprehensively listed the attacks which have been studied in literature in [53] and have also discussed the existing protocols in the field of secure ad hoc routing like: Ariadne [14] which deals with authentication of routing messages by following three different approaches; Secure Efficient Ad hoc Distance vector routing protocol (SEAD) [21] which uses one way hash chains as opposed to asymmetric cryptographic primitives employed in the other protocols; Secure Routing Protocol (SRP), which is based on DSR [13]; Authenticated Routing for Ad hoc Networks (ARAN) [16] and Secure Ad Hoc On-demand Distance Vector (SAODV) routing protocol [15], which is the secure variant of AODV. Secure Link-State Protocol (SLSP), which secures link state routing, has also been discussed. These protocols have been proposed to secure either the topology based routing protocols or source routing protocols [13].

Papadimitratos and Haas have proposed SRP (Secure Routing Protocol) [22] which secures the topology discovery aspect of routing. The authors have presented a survey of the related work in the field and have also evaluated a number of attacker scenarios. Kargl et al have studied the common attack scenarios relevant to the context of mobile ad hoc networks [58]. They have proposed the use of attack trees, suggested in [26] in order to classify the attacks. The authors have also discussed the work done in the field of authentication and key management, secure routing and mobile/distributed intrusion systems.

Chen et al have proposed Secure and Efficent Link State Routing Protocol for Ad hoc networks (SELRAN) [23] which is a secure protocol for link state routing and involves Secure Neighbor Establishment Procedure (SNEP), a secure neighbor discovery protocol and Secure Link State Update Procedure (SLSUP), a protocol to ensure the security of link state updates. Another secure protocol, which deals with the issue of preserving the anonymity, is a secure disributed Anonymous Routing protocol based on Mobile Agent (ARMA) [18]. The protocol also deals with preventing the malicious behavior of nodes during two way conversations. Secure Dynamic MANET On-demand Routing Protocol (SEDYMO) [17] secures the multihop ad hoc

routing protocol using digital signatures and hash chains. The authors highlight that, *'the secure protocol overhead is not a critical factor compared to the high network interface cost.'* The authors also classify the attacks as *Impersonation attacks, Denial of Service attacks* and *Disclosure attacks* and review the solutions related to the same. The attacks which can be mounted on the system at the link layer level have been studied in [30].

Buttyán and Hubaux have described in detail the different types of attacks on ad hoc routing protocols, along with the attack mechanisms and motives [5]. The authors have also described various routing protocols which have been proposed for secure routing in ad hoc networks. Buttyán and Vajda have proposed the concept of *Provable Security Analysis* in [19] by means of constructing an ideal world model and real world model and mapping corresponding adversaries in both, in order to verify the secure protocols. An updated version of the same [20] by Ács , Buttyán and Vajda deals with a more powerful adversary model and also presents new attacks on existing secure protocols. The list of secure protocols is not comprehensive.

2.4.2 Sensor Networks

Various protocols have also been studied specifically in the context of resource constrained sensor networks. Karlof and Wagner have studied the common attacks in the context of sensor networks and have also discussed new attacks like *Sinkhole* and *Hello Flood* in [54]. In addition, the authors have studied the countermeasures which can be used in sensor networks. Shi and Perrig have given a survey of the work done in the field of designing sensor networks in [57], where they have classified the attacks according to the security requirements and have discussed countermeasures for some of these as well.

Harsch et al have proposed a secure routing protocol for Vehicular ad hoc networks (VANETs) in [24] in which they have described the various attacks VANETs are susceptible to and have also provided countermeasures like cryptographic protection and plausibility checks to counter the same. Digital signatures and timestamps are also employed.

However, to the best of our knowledge, the field of securing the potential based approach has not yet been addressed.

2.4.3 Attacks on Routing Protocols

Various attacks on ad hoc routing protocols have been studied in literature. Some of these attacks have been mentioned alongwith references to their sources in Chapter 5.

2.4.4 Proposed countermeasures in literature

In response to the attacks on ad hoc routing protocols, various countermeasures have been proposed. Some of these are used by the secure protocols discussed in Section 2.4.1. The countermeasures have been discussed in detail in Section 6.2 where their applicability to field theoretic approach is also studied.

2.5 Summary

In this chapter, we have discussed certain concepts which are fundamental to the understanding of the field theoretic approach. Firstly, we have discussed the concept of routing and given a brief overview of the routing approaches in mobile ad hoc networks. Next we have given a brief review of the work which has been done in the field of securing ad hoc routing protocols. The work done in the field of securing sensor networks has also been discussed.

In the next chapter, we discuss the basic concept of the field theoretic approach.

Chapter 3

Basic Overview of Field Based Routing

In this chapter, we give a basic overview of the field theoretic approach. The prominent features of the routing approach are described. We identify three distinct aspects of field based approach and explain the exchange of messages during these phases.

3.1 Overview

The basic idea behind field based routing, as discussed in [6], is that there exists a scalar field around each destination node which decreases relative to distance. The potential function is unique to the scenario in which the field based concept is being utilized. Our definition of potential is discrete, unlike in Physics where the function is continuous. Each node is assigned a potential or field value and all nodes keep track of the potential values of their neighboring nodes. Access points and base stations set their potential to infinity and nodes which are closer to an access point have a higher field value. The potential field is calculated by creating the (linear) superposition of the potentials of the individual nodes of the group. Data packets are routed along the steepest gradient of the destination's field which is determined by comparing the potential values of the neighboring nodes. Direction of the steepest gradient is towards the neighbor with the highest potential value. Packets following this steepest gradient will then be delivered to any node of the (anycast) group of destination nodes. The network distance can be any metric as long as the distance function is strictly monotonously decreasing. There are basically two aspects or phases of field based routing:

- Field set up
- Packet Forwarding

Local messages are exchanged between the nodes that serve a dual purpose: first, it indicates to a node if the link to any of its neighbors is still intact or that it has broken; this helps in detecting a node failure. Second, the nodes can exchange their potential values by means of these messages.

Every node receives a message from all of its neighbors and its routing table has entries which show the potential corresponding to every neighboring node for each type of destination group. When a node needs to send a data packet, it checks its routing table and forwards the packet to the neighbor with the highest potential. If one of the links is broken or unavailable due to, for example, fading or interference, FBR can cope with that. It manages to deliver packets successfully as long as one positive gradient exists between source and destination.

3.2 Description

Field based routing is a proactive routing protocol which is inherently different from the topology based, geographical and hierarchy based protocols which have been proposed in the past, as it uses gradients in order to establish a routing state. Many secure protocols have been proposed in the past. For instance Ariadne [14], SAODV [15], ARAN [16] and so forth, which concentrate on securing either the source routing protocol or topology based approach. In topology based protocols, the nodes either exchange distance information to destination between neighbors (distance vector) or information about the complete set of links that exist in the network (link state). Even when these protocols are used in anycast routing, they rely on the same techniques as used in unicast routing, that is, routing to the closest group member over the shortest path.

The field theoretic approach fundamentally incorporates a trade off between proximity based and density based routing [4]; hence it is secure against some attacks which the above mentioned protocols encounter. Since FBR is proactive, so there is no explicit route discovery message which is circulated in the network. However due to the presence of the critical phase of establishment of potential field and the need for cooperation of neighboring nodes in the establishment of a field, there are certain attacks which are specific to the field based approach and are essentially non trivial from the security point of view. For instance, countermeasures have been proposed for preventing the modification of hop count in the ad hoc routing protocols. However with FBR, the potential values can also be manipulated, apart from the hop count. The establishment of a correct routing state is critical to the functioning of the field based routing protocol. The field based approach differs from the topology based approach because of the way the routing state is established.

The basic idea behind the field based approach is that a scalar potential field is created around every destination node and each node in the network is assigned a potential. From the messages each node receives from its neighbors, it determines the neighbor with the highest potential. This gives the steepest gradient of a field. Finally packets are forwarded along the steepest gradient. An important point here is that even though every node calculates its potential value independently of other nodes, it relies on the messages received from the neighboring nodes to determine its potential as well as to determine the steepest gradient.

Based on these principles, from a security point of view, there are three different aspects [4] of field theoretic approach that need to be secured:

- Establishment of potential field
- Gradient determination
- Packet forwarding

A number of attacks can be mounted on the first two aspects: potential field establishment and gradient determination. These attacks are specific to FBR because the manner in which the adversary can manipulate the routing state in the context of field based approach, is different from the approach in the topology based or other protocols. For instance, changing of a hop count in case of a reactive protocol and manipulation of a potential value in case of FBR would lead to the same result, that is, an incorrect routing state; however the same means cannot be used to secure them. In field based routing, potential verification is more complicated than verification of malicious hop count modification as securing the hop count does not ensure the potential verification. Hop count exists in both types of protocols. There are a number of attacks which can be mounted on the routing protocol by manipulation of the potential values as well as the hop count, which will be discussed later.

For the purpose of studying the adversary models, we need to study packets during the different phases separately: control packets and data packets. We call packets which are sent by the destination/anycast group members indicating their potential to be at infinity and the packets used for exchange of potential values as control packets as they are used to establish the potential field. These can be global broadcast messages, as in [1] or locally exchanged messages [3]. For the purpose of discussion, we consider the broadcast mechanism.

3.2.1 Establishment of potential field

The scalar potential field around each destination node in the network is created in a distributed way (for unicast, each group is considered to have only one instance). In the first phase, the destination group nodes send messages which inform the other nodes about:

- The potential of destination nodes to be at infinity
- The distance of the intermediate nodes from the destination nodes

Every node in the network must know its distance from the group members. So the group members periodically flood the network with a message indicating the anycast group they belong to and their identity. The messages contain a TTL value to limit their flooding scope, as well as to ensure the circulation of updated information in the network. Also it allows every receiver to determine its distance to the group member who initiated the flooding [4]. By listening to these messages, each node calculates its potential value according to the potential calculation function. The interval at which such messages are broadcast is decided by making a trade off between accuracy and protocol overhead [4].

3.2.2 Gradient determination

During the gradient determination, each node receives messages from its neighbors. To determine the steepest gradient of a field the nodes must know the potential values of their neighboring nodes. For this purpose they exchange their potential values. These are one hop broadcast packets and include a list of all known anycast groups and the corresponding potential value for each group, that is, Nodes exchange their potential values via local broadcasts. Link and node failures are detected by absence of several ACK packets [2].

3.2.3 Packet forwarding

Packet forwarding is done according to the steepest gradient determined in the previous step. A node forwards the packet to the neighbor with the highest potential. If the node with the greatest potential value is unreachable, the packet is forwarded to the neighbor with the next highest potential and so on until the there are no neighbors left with a higher value than its own. Nodes are not allowed to forward to a neighbor with a lower potential value to ensure that routing actually converges and loops do not form.

3.3 Summary

In this chapter, we briefly described the field theoretic approach. We identified three distinct aspects relevant from the point of view of securing the field theoretic approach which are, namely, establishment of potential field, gradient determination and packet forwarding. In the next chapter, we briefly describe the system model, which lists the basic assumptions and gives the protocol description, which shall be used as the basis for the following chapters.

Chapter 4

System Model

In this chapter, we describe the system model we shall use for describing the protocol. We list various assumptions and the protocol description, which shall be used as a basis for the following chapters. The message fields as proposed in the basic field theoretic approach will also be discussed.

In our system, we focus on a set of nodes which are operating in an ad hoc mode and communicate via the wireless medium. Each node is equipped with a radio transceiving module, a clock and processing and storage units. A communication link is established between two links if and only if they are within the transmission range of each other. The nodes' clocks are assumed to be loosely synchronized in the order of seconds, for example, by means of GPS. In case of use of public key cryptography by the nodes, we assume that each node is capable of performing public key cryptography and can store few megabytes of data. A trusted Certification Authority (CA) is present in the system that issues public key certificates binding node entities and their public keys. In case of use of shared secret keys, the keys are distributed in the initialization phase. All nodes have internal counters which are updated at the end of a particular time slot, which is described in the next section.

Key Establishment and management The problem of key setup has been studied in great detail in literature and a number of solutions have been presented to solve the issue. Some of these include key establishment using random key pre-distribution, establishing security associations, public key approach and symmetric key approach. A specific scheme for sensor networks has been suggested in [52]. Other techniques include using shared group keys or using threshold cryptography, that is, M out of N secret sharing techniques. Another approach to solving the problem can be to establish a web of trust, similar to the concept of Transitive trust and PGP trust graphs or establishing a Certification Authority (CA) and only trusting keys from several CAs. A detailed analysis of the schemes has been given in [5].

We assume that the nodes use a field based routing approach to route the packets in the network, in which the packets are delivered to the neighbor having the highest potential value, the computation procedure for which has been described later. We further assume that honest nodes in the network are willing to cooperate and relay packets for each other.

4.1 Protocol Description

For simplicity, we divide the entire duration of the time for which the network is operational into time slots of duration T. We assume that all potential maintenance messages that are transmitted between nodes regarding the potential values are done with the interval T and the network converges in every time slot of duration T.

During the interval say T_2, the existing routing table which was formed after exchange of messages in interval T_1 is used for communication and the potential values are recorded for use during the next interval. At the end of the interval T_2, all messages required for recalculation of potentials are assumed to have been exchanged and the updated routing table entries are used. During the time interval T, each node receives minimum 1 message from every neighbor stating the potential value of the neighboring nodes. It computes its potential in the following way, which has also been shown in Figures 4.1 and 4.2:

- A function f(·) which takes as its input the messages a node received from its neighbors during interval T_j and the function gives as an output a message which contains the potential value of the given node, to be used for interval T_{j+1}. We describe a general function here. The choice of the function is specific to the application scenario in which FBR is used.
- During time slot say T_1, every node computes its potential value. Initially the sources send a message indicating their potential value to be at infinity to all the nodes. The nodes calculate their potentials on the basis of the messages and transmit messages with their potential values to their neighbors.
- These packets serve two purposes:
  - Firstly, they are used as "hello" messages to indicate the current neighborhood nodes, thus if a node fails to receive a packet from a neighbor for a predefined amount of time, the neighbor is assumed to be gone.
  - Secondly, the packets are used to exchange local potential values. A node always knows the potential values of all its neighbors as required to forward messages.

  There are basically two types of packets, briefly discussed in Section 3.2, which are forwarded in the network:
  - Control packets which contain the potential values and aid in the building up of the routing tables. Potential maintenance packets are exchanged between the nodes in order to set up/update the potentials.
  - Data packets

The nodes use the updated routing tables only during the next time slot. Thus if the node with the highest potential disappears, the nodes will not use another table but forward the packet to the node with the second highest potential. If no neighbor with a potential $\geq$ the node's own potential exists, the node stores the packet for transmission in the next slot. Caching of the packet can be done to prevent packet loss. The duration of the time interval T can accordingly be set to prevent excessive data loss and, at the same time, not increase the routing overhead.

4.2 Message Fields

The messages exchanged between the nodes have the following fields:

- Sender ID: The sender node includes its ID in the message
- Destination ID: The sender node also includes the ID of the destination node in the message.
- Sequence Numbers: A number that is incremented each time the source re-advertises or broadcasts its potential value. This is used to detect if the message is new, obsolete, or is a duplicate packet delivered over an alternate path.
- Messages also have TTL fields set by the source to limit the flooding scope of its potential advertisement.
- The message also contain a Hop count field, which is a mutable field, as it is modified by intermediate nodes along a route.

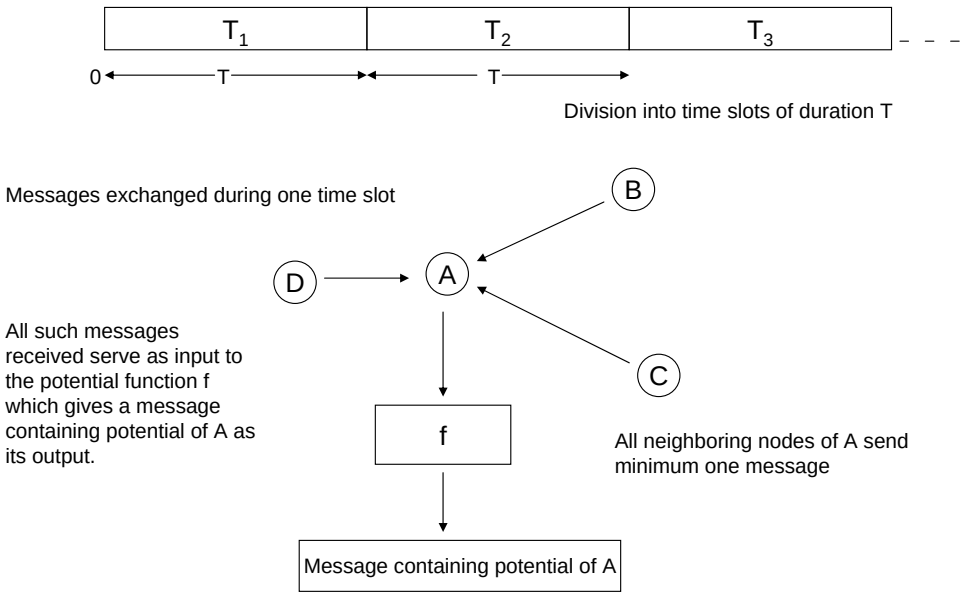

Figure 4.1: Protocol Description

During time interval T, each node receives minimum 1 message from every neighbor stating the potential value of the neighboring nodes. It computes its potential by using a function f(·) which takes as its input the messages a node received from its neighbors during interval T_j and the function gives as an output a message which contains the potential value of the given node.

During T_2 , A routes packets using the routing tables constructed during T_1 by exchange of messages as shown in the previous fig.

B	P1
C	P2
D	P3

Constructed by A during T_1

When the messages are circulated during T_2 and A receives messages with potential values of its neighbors, it constructs the updated table to be used for routing in the next interval T_3

NODE	Values used for routing during current time slot T_2	Values used for routing during next time slot T_3
B	P1	P1*
C	P2	P2*
D	P3	P3*

Figure 4.2: Protocol Description

The routing tables for the next time slot are constructed by the exchange of messages in the current time slot. The updated tables are used in the next time slot.

Sender ID	Dest ID	Hop Count	TTL	Seq No.

Figure 4.3: Message Fields
SenderID and DestinationID are included in the message. Sequence number is incremented every time the source re-advertises its potential and the TTL field limits the flooding scope of a message. Hop count is mutable field.

- So a message transmitted under the field based routing approach is of the form: <SenderID, DestinationID, Hop count, TTL, Sequence Number>, as shown in Figure 4.3.

4.3 Summary

In this chapter, we presented the system model, in which we listed the basic assumptions we make and the protocol description, which shall be used for the following chapters. We also discussed the fields present in a typical routing message according to the protocol description. In the next chapter, we discuss the attacker model and list the various attacks which can be mounted on the protocol which has been described in this chapter.

Chapter 5

Attacker Model

We define an attacker model consisting of two main attacker classes, passive and active [14]. The passive attacker does not send messages; it only eavesdrops on the network. Passive attackers are mainly threats against the privacy or anonymity of communication, rather than against the functioning of the network or its routing protocol. An active attacker injects packets into the network (but it eavesdrops as well) [14]. We characterize an active attacker, based on the information available to her, as an *Outsider attacker* or an *Insider attacker* [32]. We make an assumption that, in the absence of an attacker and provided that neither the network topology nor the potential of the nodes do change, a packet should eventually reach its destination, that is, no local maxima.

In this chapter, we describe the attacks which can be mounted on ad hoc routing protocols in general. We give the classification of an attacker being an *Outsider attacker* or an *Insider attacker*, based on her access to the network resources or the keys available to her. Next, we describe attacks which can be mounted on the field theoretic approach and provide a novel taxonomy of classifying the attacker classes.

5.1 Security Objectives

- The packet should eventually reach its destination
- The routing tables should be accurate.

5.2 General Attacks on Ad-hoc Routing Protocols

Routing is a fundamental service in any kind of network, this makes it an ideal target for attacks. The nodes in a network are generally not physically protected, hence the adversary can capture some nodes and compromise these. Some of the attacks are mounted by misbehaving nodes, for example, selfish nodes which aim at saving their own resources. Most of the routing protocols assume a trusted environment. However this may not hold true at all times. The nodes depend on the cooperation of their neighboring nodes' in order to transmit data packets or provide or access services in a network. In the absence of a centralized authority, the detection of an attack on the system or keeping a check on the nodes' behavior subsequently becomes more complicated.

Incentives While mounting these attacks, the attacker's incentives can be routing disruption, resource consumption, or the attacker might act selfish and attempt to save her own resources. By eavesdropping, an adversary might collect vital information about the network traffic. The incentive can be increasing adversarial control over the network or degrading Quality of Service (QoS) as well [5].

The attacks which can be mounted on the system can be broadly classified into three different types [17]: *Impersonation, Denial of service attacks and Disclosure attacks.*

- Impersonation: By masquerading as another node, a malicious node can impersonate a legitimate node, for example, spoofing attack.
- Denial of service attack: Crash or congest a resource. The attacker's goal is to interfere with the routing and disrupt the process. Denial of service attacks on routing protocols for ad hoc networks generally fall into one of the two categories [14], which are:
  a) Routing disruption attacks. The attacker attempts to cause legitimate data packets to be routed in dysfunctional ways, for example, wormhole, black hole, gray hole, selfish behavior, routing loop and rushing attack.
  b) Resource consumption attacks. The attacker injects packets into the network in an attempt to consume valuable network resources such as bandwidth or to consume node resources such as memory (storage) or computation power.
- Disclosure: Routing control information, node location, private or secret keys and other such private information, can be revealed to unauthorized nodes. Location disclosure attacks and passive attacks are examples of attacks of this nature. Disclosure attacks are possible since no confidentiality means are defined. The relative position of one node with respect to the other nodes of the network can be discovered.

There are several attacks in general which can be mounted on an ad hoc routing protocol. Some of these are:

- The adversary can generate false updates in the network which consumes resources and also disseminates false routing information.
- An attacker might attempt to make a route through itself appear longer by adding virtual nodes to the route; we call this attack a *gratuitous detour* [53] because a shorter route exists and would otherwise have been used.
- An attacker also might attempt to cause a node to use a route detour (suboptimal routes). This is essentially an attack against neighbor discovery.
- The attacker can affect the neighbor discovery in the following ways:
  - She can prevent two neighbors from discovering each other by jamming the network.
  - She can install wormholes [33].
  - She can create a neighbor relationship between faraway nodes. This can be achieved by tunneling control packets or by the use of wormholes.
    * By installing a wormhole, she can monitor the traffic or drop packets (Denial of Service).
- A rushing attack [29] is a malicious attack which uses duplicate suppressions at each node. The adversary floods the network with messages which leads to suppression of the later legitimate packets that reach the nodes.
- Other routing disruption attacks which the adversary can mount on the system are of the form:
  - Creation of a routing loop, causing messages to be routed in a cycle without reaching their actual destination, leading to available bandwidth consumption.
  - By dropping received messages: Black hole/ gray hole depending on whether the attacker drops all messages or drops messages selectively.
  - The attacker can partition the network by injecting forged routing packets which interferes with the communication between two nodes and prevents the messages sent by one node from reaching another.

  - The attacker can also mount attacks that involve injecting messages in the network which may lead to resource consumption:
    * Inject extra data packets in the network
    * Inject extra control packets in the network

- The attacker can modify the messages.
- The attacker can also replay old messages. She can modify one or more fields and then replay the messages in the network.
- An outsider attacker (and insider attacker as well) can inject erroneous information, replay previous routing messages or modify valid routing info.
- Modification of mutable information in the routing messages, for example, hop count.
- Denial of Service (DoS) attacks can be mounted on the network by the adversary in the following ways:
  - Flooding a network thereby preventing legitimate network traffic from reaching its destination nodes.
  - Sending messages to a particular node more than it can possibly handle, thereby preventing access to a source.
  - Preventing a particular node from sending messages or accessing a particular service implemented by a source.
  - Disrupting services to a particular node.
- The attacker can also create a *sinkhole* [54] where the attacker's goal is to lure the traffic towards a particular compromised node. The attacker can later drop the packets or modify and forward them. A *sinkhole* can be created with the adversary at the center [54].
- Another attack which can be mounted on the system is the *Hello Flood* [54], in which false neighbor relations are created by the attacker.
- An attacker can mount a *Sybil attack* on the network [27], where the attacker node claims multiple identities. A sybil attack can be of various types (geographical, at data link layer etc.) and is a threat at every layer [28].
- An attacker can also mount a node replication attack [5] where a number of nodes share the same identity.
- Another type of attack which the adversary can mount on the system is the *Acknowledgment Spoofing* [54] wherein the adversary can falsely make a weak link seem stronger and cause loss of packets by injecting false acknowledgments.
- There can be attacks on the link layer as well, for instance when an adversary resorts to sending more frames than its allocated share, which could affect the upper network layer and disrupt the routing protocol operation [30].
- There can be various attacks related to node positioning [5], for instance node displacement, wormhole, malicious distance enlargement, dissemination of false position and distance information.
- Another type of DoS attack which has been proposed in [18] is the Malicious ID attack. An intermediate node is compromised or adverse and provides forged ID or spoofing address to the source node in the path discovery phase. If there is one or more malicious node(s) that provide(s) the false ID(s), this will result in invalid routes discovered by the source node. *The compromised intermediate node always attempts to provide as many malicious IDs as possible, and thus establishes as many invalid routes as possible as well [18].*

5.3 Outsider Attacker

We assume an omnipresent but computationally bounded attacker. However the attacker is not an authenticated or authorized member of the network. An outsider does not possess the cryptographic keys required to generate genuine messages and disrupt the routing process by disseminating forged routing information. An outsider attacker can insert fake messages that lead to resource consumption due to the verifications involved, however in the absence of the legitimate cryptographic primitives, an attacker can not generate messages which can be used in the determination of the steepest gradient. There are mainly two attacker classes [14], passive and active. As discussed in [14], the passive attacker can easily eavesdrop on the communications between the nodes on the network but mainly violates the objectives of privacy and confidentiality in a network. On the other hand, an active attacker inserts arbitrary messages in the network and eavesdrops as well. The adversary can also alter or spoof packets, to infringe on the authenticity of communication or inject interfering wireless signals to jam the network. Another form of outsider attack is to disable the nodes. To this end, an attacker can inject useless packets to drain the receiver's battery; she may physically destroy nodes.

5.4 Insider attacker

An insider attacker has compromised one or more nodes in the network and consequently the network is susceptible to more attacks. The insider attacker, as opposed to an outsider attacker, has access to the set of cryptographic keys used for encrypting and signing the messages. She can use the primitives to generate legitimate messages or to modify the immutable fields in the messages circulated in the network. The attacker can generate the messages used to determine the steepest gradient and also has access to the immutable fields in the messages circulated in the initial phase of field set up as well as during packet forwarding; hence it poses a greater threat to the routing protocol as compared to the outsider adversary. For instance, the adversary can manipulate the hop count in the potential field messages and disseminate wrong routing information in the network. The adversary can also inject packets into the network solely with the aim of resource consumption.

5.4.1 Attacks which can be mounted on field based approach

We classify the insider attacker into three classes, namely:

- Single adversary/multiple non colluding non neighboring adversaries: This class of attackers includes a single adversarial node in the network or multiple non colluding adversarial nodes which are not neighbors and do not operate together.

- Multiple neighboring adversarial nodes/ chain of adversarial nodes: This class of attackers includes two or more adversarial nodes which are neighbors or exist in a chain. An tunnel is not possible in this case; however the adversarial nodes are in a chain or are neighboring.

- Multiple colluding nodes: This class of attackers includes multiple colluding adversarial nodes which may or may not be neighbors. Tunneling of messages between various attacker nodes is possible.

Possible Attacks There are a number of attacks which can be mounted on either of the field set up phases or the packet forwarding phase as has been described in Section 5.2. We describe the different attacker classes and the attacks that can be mounted on the three phases by the attacker. Additionally, some of the attacks discussed in Section 5.2 have been considered with regards to their impact on the field theoretic approach specifically.

5.4.2 Single adversary/multiple non colluding non neighboring adversaries

We now describe the attacks that an adversary in this class can mount on each of the three phases, as described in Section 3.2. The different attack scenarios have been discussed in detail. In the next chapter, we describe how the security primitives can be used to counter some of these attacks.

Potential field establishment

The purpose of the messages during this phase is to broadcast the distance from destination nodes, let the correct and accurate information be transmitted or flooded efficiently throughout the network.

Aim The adversary's aim is to attack any of three aspects: She can cause wrong distance information to be disseminated, can interfere with the flooding or can create false neighbor relations, that is, cause two neighbors to not discover each other or introduce wrong neighborhoods.

Malicious modification of hop count The only mutable field in the potential field message is the hop count and by manipulating the hop count, the adversary can mount any of the following attacks: The adversary can receive the messages in the initial phase and modify and forward them, leading to dissemination of incorrect routing information or in some cases, can cause the network to be partitioned. In Figure 5.1, the propagation of control packets is shown in the absence of an adversary in the network. In Figure 5.2, three different ways in which the hop count can be manipulated by the adversary have been depicted (in the same scenario as in Figure 5.1), which are:

- *Decrease:* When the potential field is being established, the adversary can *lower the hop count* in one or more messages, indicating to be closer to the destination and causing packets to be routed towards it. These are route disruption attacks [5] as the manipulation of control packets prevents the creation of legitimate routes. In Figure 5.2, the adversary decreases the hop count by 1 and forwards the message to node C. Not only would Node C discard the accurate messages with the correct hop count, but it would also calculate a wrong potential value and propagate the same through the network. All subsequent messages from Node C would be routed towards the adversary, eventually creating a sinkhole.
- *Increase:* The adversary can make a route through itself appear longer, gratuitous detour [53] by *increasing the hop count* by more than 1 and forwarding the message to its neighbors. It would lead to unnecessary resource consumption. She can also mount a DoS attack on an honest node, as the node sees itself as being farther from the destination or the sink than it actually is, again creating a wrong routing state. For instance, in Figure 5.2, the adversary is at a distance 2 from the sink, however it increases the hop count to 5 and forwards the message to Node F. This node would consider itself it to be at a hop count 5 from the neighbor even though the actual distance may be less. A selfish node can also deliberately increase the hop count in order to save its own resources, for instance, in Figure 5.2, Node F would use the route passing through I, as the adversarial node advertises a greater value of hop count to save its own resources.
- *Same Distance Fraud:* The attacker can also mount an attack by *not changing the hop count* and forwarding the message. It would again lead to wrong distance estimation as the nodes would presume that they are closer to the sink than they actually are. The attacker can send the message back to the sender node as well. In Figure 5.2, the adversary forwards the message to Node G without incrementing the hop count. As a consequence, Node G would consider itself to be at a distance 2 from the sink whereas the actual distance is 3. The accurate hop count denoted by the message from honest node H would be discarded and Node G would choose the message from the adversarial

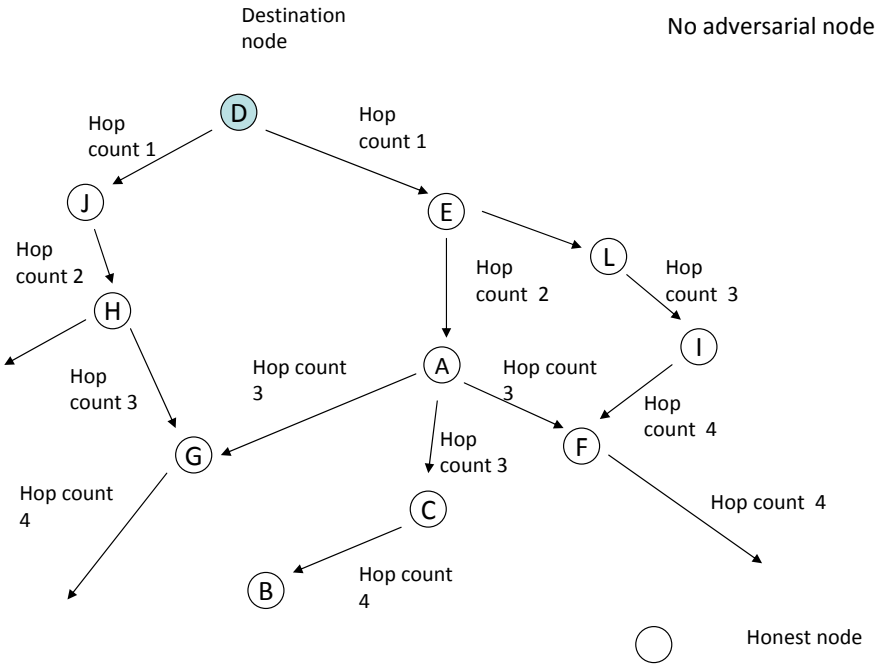

Figure 5.1: Scenario: In the absence of an adversary

node in order to calculate its potential value. All messages from G would be routed towards the adversary.

- *Consequences:* Most of these attacks can be mounted by the adversary with the aim of creating a sinkhole, wherein all the data packets would be routed towards it by virtue of its false higher potential. The attacker can subsequently manipulate the packets in various ways; she can drop all the data packets, creating a black hole, she can selectively drop packets, creating a gray hole or she can change the hop count and replay the messages at a later stage.

Message Replay The adversary can replay a message in the same timeslot as the one for which the message was intended or in a later timeslot. The adversary can cache the message and even modify it, before replaying it at a later time. Message modification would lead to incorrect routing traffic in the network. In Figure 5.2, the adversary can cache the message with hop count 1 from Node E. She can replay the same message in another time slot, even after the topology would have changed and forward the message to its neighbors, similar to the message forwarded to Node G in the figure.

Sybil attack It is possible to mount a sybil attack [27] where the attacker assumes multiple identities and creates a wrong routing state in the network. The attacker can then create a sinkhole in the network. A similar type of attack can be mounted on the system where a single node can claim to have various potentials or IDs with the consequence that an incorrect routing state is created. The adversary can also launch a geographical sybil attack [24] where she claims to be at multiple positions. For instance, in Figure 5.3, the adversary creates three sybil node identities as shown. Each identity can assume to be a different distance from the node or at a different potential. By indicating a lower hop count or a higher potential value, the adversary causes all packets to be routed towards it and inaccurate information to be propagated in the network by the honest nodes receiving the message from the sybil node identities.

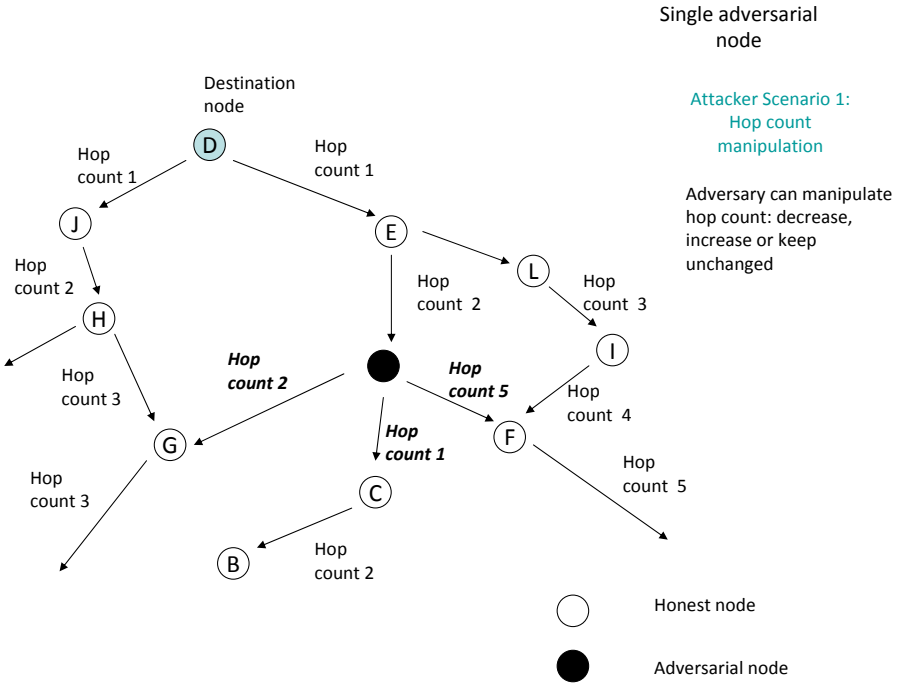

Figure 5.2: Attacker scenario 1: Hop count modification

Three different ways in which the hop count can be manipulated by the adversary: The adversary is at a distance 2 from the sink. however the adversary decreases the hop count by 1 and forwards the message to node C. Not only would Node C discard the accurate messages with the correct hop count, but it would also calculate a wrong potential value and propagate the same through the network. All subsequent messages from Node C would be routed towards the adversary, eventually creating a sinkhole. Another attack is mounted by the adversary as it increases the hop count to 5 and forwards the message to Node F. This node would consider itself it to be at a hop count 5 from the neighbor even though the actual distance may be less. Hence, Node F would use the route passing through I, as the adversarial node advertises a greater value of hop count to save its own resources. As another form of malicious modification, the adversary forwards the message to Node G without incrementing the hop count. As a consequence, Node G would consider itself to be at a distance 2 from the sink whereas the actual distance is 3. The accurate hop count denoted by the message from honest node H would be discarded and Node G would choose the message from the adversarial node in order to calculate its potential value. All messages from G would be routed towards the adversary.

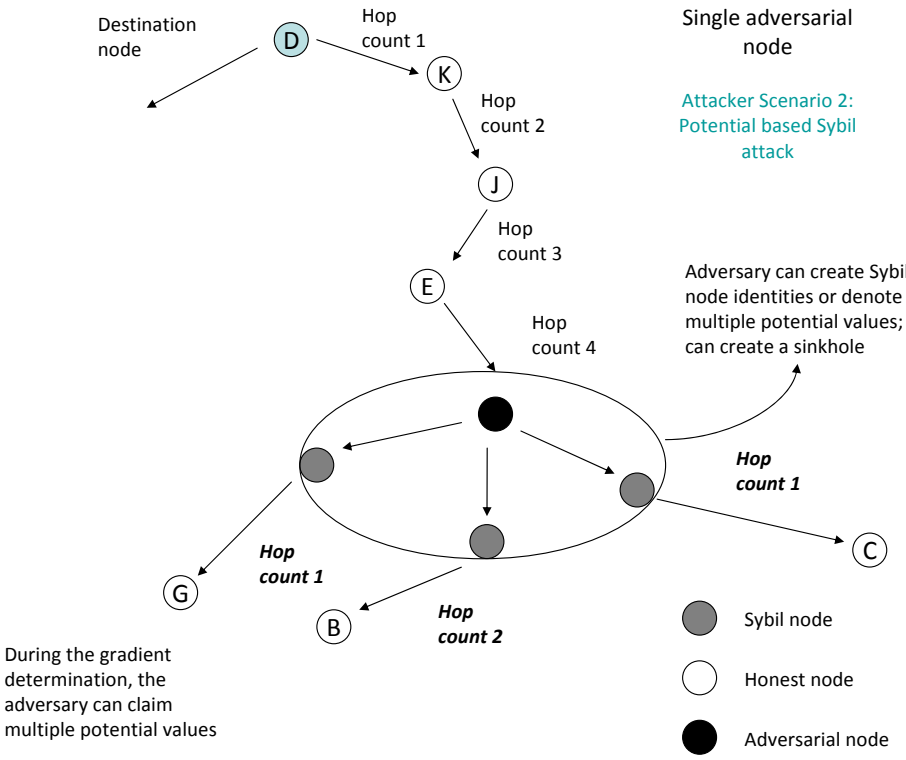

Figure 5.3: Attacker scenario 2: Potential based Sybil

The adversary creates three sybil node identities as shown. Each identity can assume to be a different distance from the node or at a different potential. By indicating a lower hop count or a higher potential value, the adversary causes all packets to be routed towards it and inaccurate information to be propagated in the network by the honest nodes receiving the message from the sybil node identities.

The attacker can also indicate a different position than its actual, that is, a false position advertisement [24].

Sink Impersonation In case of the potential field establishment, the adversary can create messages indicating to be at potential infinity and then subsequently drop all the data packets or perform other operations as indicated above, with the intention of increasing the adversarial control over the network or simply resource consumption. It would lead to a traffic imbalance in the network as well. For instance, in Figure 5.4, the adversary creates a message indicating itself to be at potential infinity. As there is no way of checking or verifying a node's potential value, the honest nodes assume the adversary to be a sink node. Node E receives two messages, one from the actual sink from which it is at a hop count 2 and from the adversary, from which it is at hop count 1. Node E would route all the packets towards the adversary and calculate a wrong potential value as well.

Denial of Service attacks Apart from these attacks there are a number of resource consumption attacks which can be mounted by the adversary on the system: In the initial phase, a denial of service attack can be mounted on the system as the adversary can forge the identity of an honest node at a lower potential than infinity and create a message indicating the honest node in the network to be at potential infinity. This would lead to flooding of the victim node with packets which are not meant to be routed through the victim node and would lead to dropping of some packets due to excessive traffic. Here the adversary exploits the fact that the cooperation of other nodes is important while establishing the potential field and the nodes have to rely on the information forwarded to them by the neighboring nodes. It would divert the routes through the victim nodes. These are route diversion attacks [5] as these attacks prevent the use of previously established routes in the network.

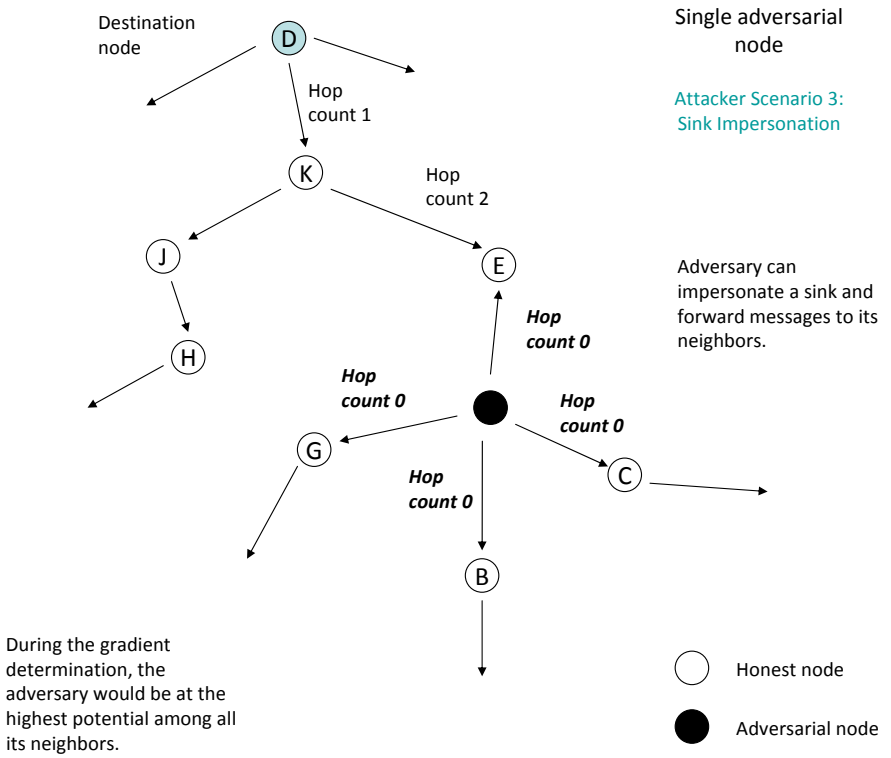

Figure 5.4: Attacker scenario 3: Sink Impersonation

The adversary creates a message indicating itself to be at potential infinity. As there is no way of checking or verifying a node's potential value, the honest nodes assume the adversary to be a sink node. Node E receives two messages, one from the actual sink from which it is at a hop count 2 and from the adversary, from which it is at hop count 1. Node E would route all the packets towards the adversary and calculate a wrong potential value as well.

Node Impersonation The adversary can also mount attacks involving forging of identity of other nodes for which she possesses the private or secret keys. The adversary can also insert messages with false updates in the network, and flood the network, leading to creation of extra control traffic in the network, which might cause the legitimate packets to be dropped.

Against neighbor discovery Apart from the above mentioned attacks which threaten to disrupt the routing, there are some attacks which can be mounted by the adversary which thwart the neighbor discovery process. These attacks are of the nature: Creation of a wormhole [5], partitioning of the network by forwarding selective packets. The adversary can also generate false messages which may lead to a state such that the network may be disconnected but the routing state falsely shows the destination as reachable.

Gradient Determination

The motive of the honest nodes in this phase is to exchange potential values with one hop neighbors. In this phase, the adversary can either be a sender or a receiver.

Adversary as a sender As a sender, the adversary can indicate a higher potential and create a sinkhole, causing all data packets to be subsequently routed towards it. She can subsequently create a black hole or a gray hole or perform other operations as described in the previous phase. However, in some cases, in order to save its own resources, the adversary can also modify a message to indicate to be at a lower potential and force the other nodes to use suboptimal routes and incorrect routing information. As mentioned in the protocol description, the nodes exchange their potential values in order to establish the steepest gradient. The adversarial node can manipulate the hop count, calculate a higher or a lower potential value. The malicious modification of hop count can be done in many ways, as described for the previous phase.

Adversary as a receiver As a receiver, the node can forward the message it gets from one of its neighbors, modify some fields and forward it to another neighbor, creating a false neighbor relationship.

Packet Forwarding

The motive of the honest nodes during this phase is to forward data packets along the steepest gradient; to the neighbor with the highest potential. This aspect of field based routing is susceptible to most of the attacks that have been studied in literature based on data packet forwarding. The adversary can mount attacks either as a sender or as an intermediate node.

Adversary as a sender As a sender, the adversary can generate data packets with another node's ID, for which she possesses the private or secret keys, thus creating incorrect routing traffic in the network which leads to resource consumption as well as disrupts the routing process. The attacker can also give the wrong potential value in the message, which increases the possibility of creation of a routing loop. The adversary can also insert messages in the network with varying results.

Adversary as intermediate node The adversary can replay a previously cached packet. An insider adversary who has compromised one or more nodes can mount such attacks on the process of data packet forwarding. Since she has the necessary cryptographic primitives and is an authenticated member of the network (as it has compromised one or more nodes), she can receive the data packets in the network and subsequently drop the packets. If all the data packets are dropped, it leads to a black hole. Dropping of specific packets creates a gray hole. Since the adversary has access to the messages circulated in the network, it can effectively partition a network by forwarding selective data packets.

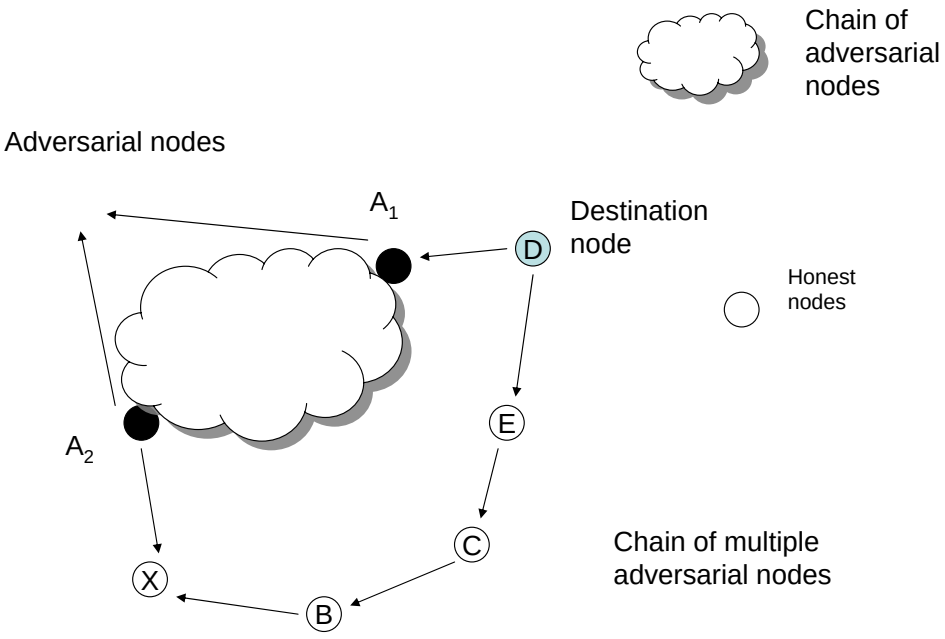

Figure 5.5: Multiple neighboring adversarial nodes/chain
A chain of multiple nodes can mount a number of attacks on the system, for instance, node X can get a message from A_2, indicating a lower hop count than in the message from B. A_1 is close to the destination node and the potential field message can be passed via the chain without incrementing the hop count, and node X would believe itself to be closer to the destination. A sinkhole is eventually created.

Data packet manipulation Since the adversary is an insider, she can even modify the packets to manipulate the hop count or any other parameters. The adversary can then mount a replay attack on the system as is possible in the case of control packets. It leads to unnecessary resource consumption.

Selfish node attacks In ad hoc networks, node resources are constrained and a selfish node would not forward data packets, to save its own resources. This would lead to use of suboptimal routes as nodes would have to send the packets to the neighbors with the next highest potential, which might be a longer route or a route through a region with less members of the destination group, thus affecting the robustness of the protocol.

5.4.3 Multiple neighboring adversarial nodes/chain

The attacks which can be mounted by an adversary belonging to this class include the attacks which can be mounted by a single adversary or multiple non colluding non neighboring nodes. A chain of multiple nodes can mount a number of attacks on the system, for instance, in Fig 5.5, node X can get a message from A_2, indicating a lower hop count than in the message from B. A_1 is closer to the destination node and the potential field message can be passed via the chain without incrementing the hop count, and node X would believe itself to be closer to the destination. Eventually, a sinkhole is created. In addition, the following attacks are possible:

Node isolation If a node or few nodes are connected to the rest of the network via a chain of colluding nodes, as shown in Fig 5.6, a number of attacks can subsequently be mounted on these nodes, for example, the potential field messages can be maliciously modified and forwarded to the isolated nodes. In the absence of other honest nodes, these nodes can only rely on the information transmitted by the adversarial nodes. The chain of nodes can create a sinkhole and mount other attacks e.g. sybil or node replication attacks and isolate the honest nodes from the rest of the network. A large number of denial of service attacks can be mounted on the system subsequently as well.

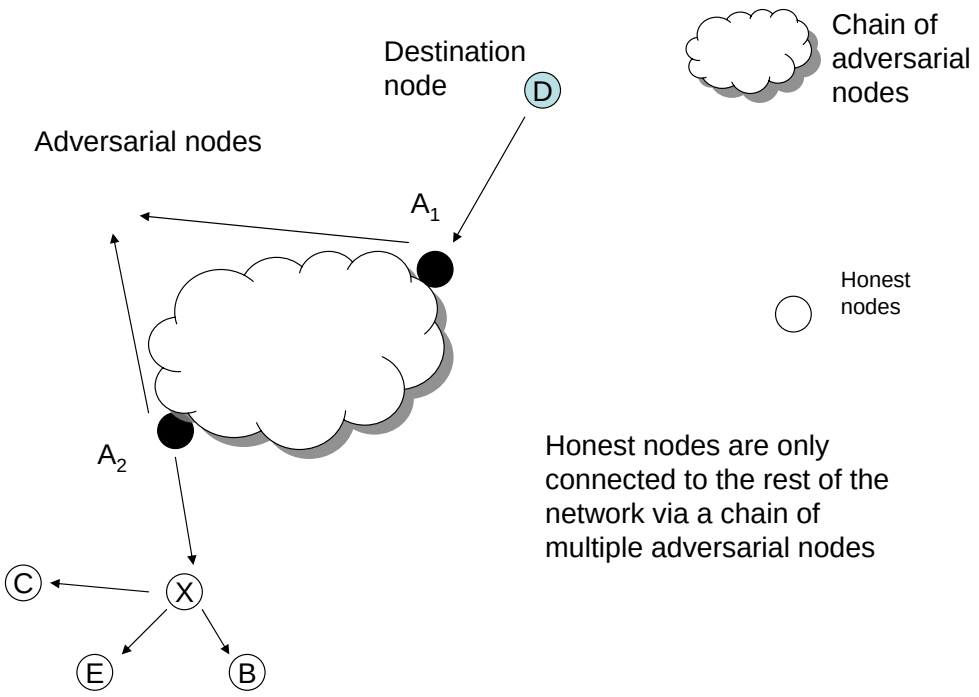

Figure 5.6: Node(s) Isolation

A chain of multiple nodes can mount a number of attacks on a set of honest nodes which are connected to the rest of the network via a chain of colluding nodes, for instance, node X can get a message from A_2, indicating a lower hop count than the actual hop count. A_1 is closer to the destination node and the potential field message can be passed via the chain without incrementing the hop count, and node X would believe itself to be closer to the destination. A sinkhole is eventually created as the potential field messages can be maliciously modified and forwarded to the isolated nodes. In the absence of other honest nodes, these nodes can only rely on the information transmitted by the adversarial nodes.

Potential Field Establishment

Using same ID Many adversarial nodes in a chain can claim to have the same ID or claim to be at the same potential. It will lead to creation of a sinkhole in the network. The attack is similar in nature to the node replication attack [5].

False potential claims A specific kind of attack related to impersonation would be if a number of colluding nodes all claim to be at potential infinity. This kind of attack exploits the fundamental density based nature of field based approach. If there are more colluding nodes claiming to be at potential infinity and away from the legitimate nodes in the network, it would lead to all packets being routed to the corrupted nodes (density based routing).

Rushing/chain attack Another type of attack that emerges as a consequence of forwarding of packets between colluding adversarial nodes in a chain is similar to the rushing attack [29]. If the adversary modifies routing information and delivers a packet, it would lead to incorrect potential calculation and subsequently to creation of a sinkhole and resource consumption.

Gradient Determination

Against potential verification A chain of adversarial nodes can manipulate their potential values. This can be done because any of the adversarial nodes receiving a message can manipulate the hop count and forward the message to its colluding node which can use the message for its potential calculation by claiming the message to have been sent by another colluding node. For two or more nodes, it would not be possible to verify the false potential values so calculated, correctly. After creating a routing state wherein any of the adversarial nodes are at a higher potential, a sinkhole can be created where all packets are sent to the adversarial chain of nodes.

Packet Forwarding

If any of the adversarial nodes has created a sinkhole, all data packets would be routed towards it.

Chain attack A chain attack can be mounted on the routing protocol where one of the colluding nodes encloses the packet and sends it to an adversary in the chain and essentially prevents the packet from reaching its destination. This may lead to resource consumption or creation of a routing loop.

5.4.4 Multiple colluding nodes

The adversary belonging to this class can mount all the attacks which are possible in the previous two attacker classes. Here, we discuss the other attacks which are possible.

Tunneling In particular, in this class of attackers, tunneling of messages between multiple colluding adversaries in different parts of the network makes the network susceptible to more attacks than in the previous two attacker classes.

Potential Field Establishment

Tunneling of potential field messages The adversary can receive the messages in the initial phase and modify and tunnel them to any other colluding node, leading to dissemination of incorrect routing information or in some cases, can cause the network to be partitioned. For instance, when the potential field is being established, the adversary can modify the hop count in one or more messages and tunnel the packets to another part of the network, thus creating an incorrect routing state. In Figure 5.7, attacker A_2 is closer to the sink than A_1. She can tunnel the potential field messages she receives to A_1 who can forward the message as if it was received from a sink. An incorrect potential field would be setup around node A_1.

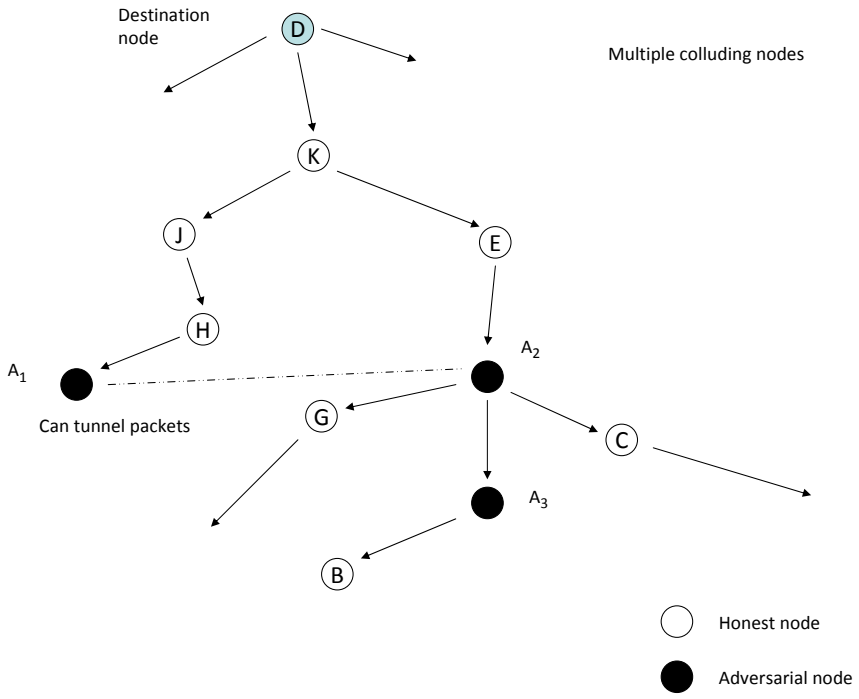

Figure 5.7: Multiple colluding nodes

Attacker A_2 is closer to the sink than A_1. She can tunnel the potential field messages she receives to A_1 who can forward the messages as if these were received from a sink. An incorrect potential field would be setup around node A_1. Also, adversarial node A_1 can forward the data packets she receives to A_2 which would transmit these packets in its vicinity. If the attackers mount the rushing attack on the system, the incorrect routing messages would reach the honest nodes before the accurate potential field messages. As a consequence, the honest nodes would calculate a wrong potential field, discarding the legitimate potential field messages received later.

Message replay The adversary can also cache the message, modify some fields, and tunnel it to another adversary who can replay the message.

Impersonation The adversary can also mount attacks involving forging of identity of other nodes. Many nodes can claim to have the same ID or claim to be at the same potential. It can lead to creation of sinkhole in the network, if the nodes are placed at specific positions in the network, as the field theoretic approach is density based. The attack is similar in nature to the node replication attack [5]. A specific kind of attack of this nature would be if a number of colluding nodes all claim to be at potential infinity. This kind of attack exploits the fundamental density based nature of field based approach. If there are more colluding nodes claiming to be at potential infinity and away from the legitimate nodes in the network, it would lead to all packets being routed to the corrupted nodes.

Rushing Another type of attack that emerges as a consequence of tunneling of packets or delivering messages via an out of band connection is *Rushing* [29]. If the adversary modifies the routing information and delivers a packet via a tunnel or an out of band connection, it would lead to incorrect potential calculation and subsequently to creation of a sinkhole and resource consumption. In Figure 5.7, if the attackers mount the rushing attack on the system, the incorrect routing messages would reach the honest nodes before the accurate potential field messages. As a consequence, the honest nodes would calculate a wrong potential field, discarding the legitimate potential field messages received later.

Gradient Determination

All attacks from the previous two attacker classes can be mounted and in particular, false neighbor relations can be created due to tunneling or out of band connection.

Packet Forwarding

An insider adversary who has compromised one or more nodes can mount attacks on the process of data packet forwarding. A tunneling attack can be mounted on the system where one of the colluding nodes encloses the packet and tunnels it to the other colluding nodes and essentially prevents the packet from reaching its destination. This may lead to resource consumption or creation of a routing loop. For instance, in Figure 5.7, adversarial node A_1 can forward the data packets she receives to A_2 who would transmit these packets in its vicinity.

Tunneling The adversary can also mount another type of attack which affects the routing state. She can receive the messages or insert new messages, which lead to the creation of local maxima.

5.5 Summary

In this chapter, we described the attacker model. We listed the major security objectives we are working towards. We briefly described the different attacks which can be mounted on the ad hoc routing protocols in general. There are mainly two classes of attackers, that is, the broad classification into outsider attacker or insider attacker, in accordance with the information available to the attacker and the type of attacks she can mount on the system. With reference to the field theoretic approach, we further classified the insider attacker into three main attacker classes: single adversary/multiple non colluding non neighboring adversaries, multiple neighboring adversarial nodes/chain of adversarial nodes and multiple colluding nodes. We described the different attacks which can be mounted by these attackers on the field theoretic approach and the impact of these attacks on the routing protocol.

In the next chapter, we discuss the measures which can be implemented in order to secure the protocol against the attacks mentioned in this chapter.

Chapter 6

Secure Protocol

In this chapter we discuss the different approaches which can be used to secure the routing protocol against the attackers as discussed in the previous chapter. Firstly, the countermeasures for securing the routing protocol against the outsider attacker are suggested. For the purpose of securing the protocol against the insider attacker, we initially discuss various security mechanisms that safeguard the protocol against specific attacks. After reviewing the available countermeasures, we propose five different approaches for securing the field set up, by combining some of the security mechanisms discussed previously. The approaches are analysed in the next chapter. We also discuss the special case of sensor networks and suggest certain efficient countermeasures, which can be used to secure the field based routing protocol in sensor networks.

6.1 Against Outsider Attacker

We propose certain mechanisms which shall secure the protocol against the attacks mounted by an outsider attacker. When a node has to send a packet to another node, an authentication mechanism is required to prevent unauthorized messages from being inserted in the network. The nodes may use any of the following mechanisms:

- Use of shared secret keys
- Use of a Public Key Infrastructure (PKI)

The advantage which the latter has over the former is that in case of public keys, key distribution is not a major issue. For shared secret keys, there has to be confidentiality when the keys are distributed. The key distribution problem has been studied in detail and a number of solutions for the same have been proposed in [5] and briefly discussed in Chapter 4. However the message overhead involved in case of a public key infrastructure is more and PKI is susceptible to DoS attacks as well. The adversary might flood the network with bogus packets for signature verification.

Origin authentication The sender node signs the message using its own private key and the value obtained is transmitted along with the message. The destination node or any intermediate node can verify that the message was sent by the sender node by verifying the digital signature. Other signature schemes or use of MACs can be considered as well. For the purpose of discussion, we assume that the nodes use a secure authentication scheme.

Additional fields We also propose the use of two more message fields in addition to the fields specified above: *TimeSlot* and *CTR*. Each node is equipped with a clock which is loosely synchronized with the other nodes' clocks. Hence every node is aware of the time slot in which the network is operating at the time it transmits the data packet. The sender node marks the number of the current time slot in the field TimeSlot. CTR or Counter is set to 0 by every node at the beginning of every new time slot. The sender node sets the field as 1 in the first data packet which it transmits in a particular time slot, the CTR field in the next packet is set as 2

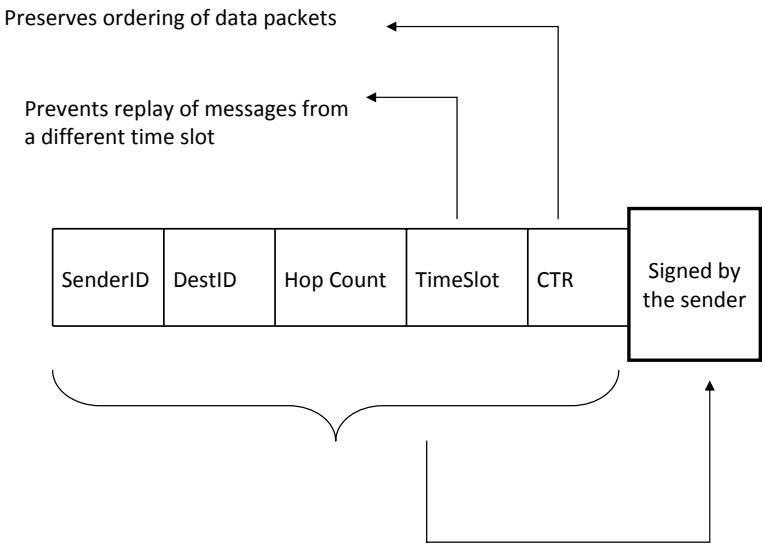

Figure 6.1: Secure Protocol
The sender node marks the number of the current time slot in the field TimeSlot. CTR or Counter is field which is set to 0 by every node at the beginning of every new time slot. The 4-tuple {SenderID, DestinationID, TimeSlot, CTR} uniquely defines a packet.

and so on. Hence the 4-tuple {SenderID, DestinationID, TimeSlot, CTR} uniquely defines a packet, as shown in Figure 6.1.

Securing neighbor discovery In order to secure the protocol against attacks thwarting neighbor discovery (e.g. wormholes), packet leashes [33] can be used. The details for the same have been mentioned in the next section.

TTL field The *TTL (Time To Live)* field is not required in the packet as the lifetime of a packet is bounded by TimeSlot and CTR field. At the beginning of a new time slot, all the packets which were meant for the previous time slots are discarded as obsolete. Within a time slot, if a packet with a higher CTR value from the same node is received, then the packet which has a lower CTR value would be discarded and the updated packet values would be used.

Sequence Numbers The *Seq No.* field is not required in the packet as the outdated information can be discarded on the basis of the information contained in the TimeSlot and CTR fields. Whenever a node sends an update during a particular time slot, it increments the CTR field as well. The CTR field is incremented with every packet which the node sends. Hence in the same time slot, a packet with a higher value of CTR would be preferred over a packet with lower CTR value as the latter would have outdated information as compared to the former. The value of TimeSlot field distinguishes between packets meant for different time slots.

6.2 Against Insider Attacker

As explained in Section 3.2, there are three aspects of field based routing, which are:

- Establishment of the potential field
- Gradient determination
- Packet forwarding

We first discuss the available security mechanisms for securing the protocol in general. Next, we propose some approaches where certain mechanisms can be used simultaneously in order to secure the protocol. The basic protocol structure in the three phases has been shown in Figures 6.2, 6.3 and 6.7.

6.2.1 Establishment of Potential field

The aim is to disseminate the information for nodes to know their distance from the destination nodes and to let the correct and accurate information be transmitted or flooded efficiently throughout the network. For the purpose of discussion, we consider the global broadcast mechanism proposed in [1]. All the destination nodes broadcast packets which indicate their potential to be at infinity and by means of the hop count, each node can determine its distance from the sink or destination nodes and calculate its own potential value.

Message fields The message has two parts: the immutable fields include {Potential, SenderID, DestinationID, TimeSlot, CTR }. The potential field is set to infinity; TimeSlot field is indicative of the slot in which the packet was sent and CTR indicates the order in which the packet was sent. The mutable field is the hop count which is incremented by every node as the message is circulated in the network as shown in Figure 6.2. The static information generated by the anycast group is signed by the sender node. Subsequent nodes sign the concatenation of the two parts, as shown in Figure 6.2.

Measures against sink impersonation We assume that all destination nodes have a certificate by a trusted authority which is also attached with the potential field messages broadcast by the destination nodes. Similarly, the sender ID is a special anycast group ID which indicates the anycast group the node belongs to. We assume that the destination nodes hold the valid certification as described and can not be compromised by an adversary.

Intermediate node operations On receiving the potential field message, the intermediate node verifies the origin of the packet with the help of the certificate and the authentication mechanism used. It verifies the authenticity and integrity of the packet. The node records its distance from the destination node, increments the hop count and uses one of the techniques mentioned in Section 6.2.1 in order to validate the change. The intermediate node signs the mutable field, that is, the hop count and the header information and forwards the packet, which includes the immutable fields originally signed by the destination node as well.

Efficient Flooding A number of efficient flooding techniques have been proposed in literature. In order to limit flooding, some protocols give priority to the packets sent through neighbors that less frequently forward packets. However this has various disadvantages as an adversary can take advantage of this scheme. A counter based mechanism has been proposed in [23]. An efficient scheme based on local transmission of messages has been described in [3]. The local approach used by the protocol helps to combine the first two phases in field set up; Potential field establishment and gradient determination and has been discussed in detail in Section 6.3.

Plausibility checks Harsch et al extended the plausibility checks presented in [48] and presented a list of plausibility checks in [24], which can be used to check the time, acceptance range and velocity of the packet. In the present context, the checks on the acceptance range can check whether the packet has traveled a distance greater than the maximum transmission distance of the communicating devices. Separate plausibility checks on time are not required as the fields TimeSlot and CTR are indicative of the time when the packet was transmitted. On reception of a packet, a node executes the different checks in sequence, and drops the packet if any of them fails.

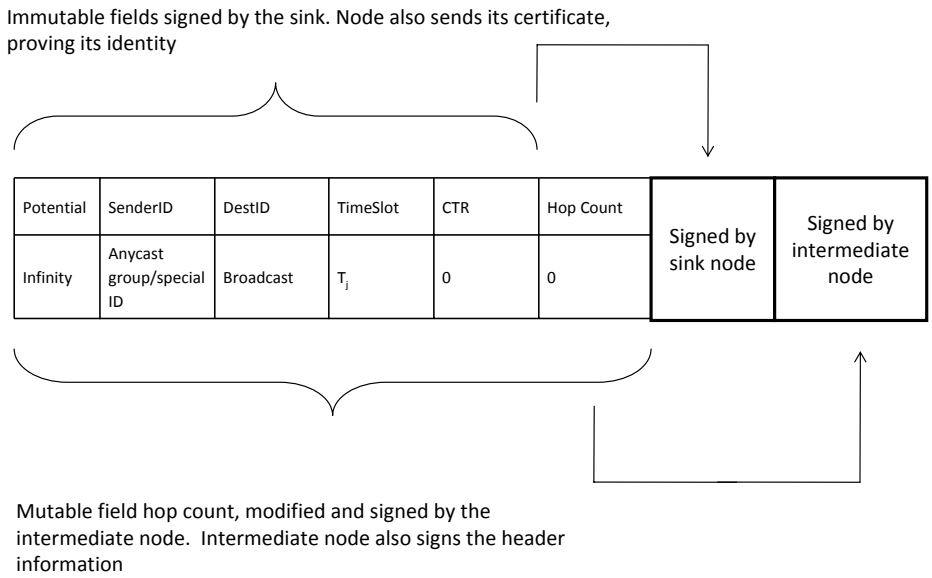

Figure 6.2:
The message has two parts: the immutable fields include {Potential, CTR, TimeSlot, SenderID, DestinationID}. The potential field is set to infinity; TimeSlot field is indicative of the slot in which the packet was sent and CTR indicates the order in which the packet was sent. The mutable field is the hop count which is incremented by every node as the message is circulated in the network.

Securing mutable information

For securing the mutable field, that is, the hop count, we can use any of the following schemes: Hash Chains, Tree authentication (Merkle trees), hash tree chains, per hop hashing scheme or the hop count can be eliminated completely.

Hash Chains One way hash chains are frequently used in the design of secure protocols ([14], [17]). A random seed is chosen and the value is successively hashed in order to create a hash chain. The elements of the hash chains are released in the reverse order. As stated in [34], *the one way chains have two important properties (assuming the hash function to be a secure cryptographically secure one way hash function):*

- Anybody can authenticate that a value v_j really belongs to the one way chain, by using an earlier value v_i of the chain and checking that $H^{j-i}(v_j)$ equals v_i, where H is a one way hash function.
- Given the latest released value v_i of a one-way chain, an adversary cannot find a later value v_j such that $H^{j-i}(v_j)$ equals v_i. Even when value v_{i+1} is released, a second pre-image collision resistant hash function prevents an adversary from finding $v_{i+1}^{'}$ different from v_{i+1} such that $H(v_{i+1})$ equals v_i. These two properties result in authentication of one-way chain values: if the current value v_i belongs to the one-way chain, and we see another value v_j with the property that $H^{j-i}(v_j)$ equals v_i, then v_j also originates from the same chain.

Tree authentication Tree authentication [34] is based on Merkle trees. Merkle trees [51] are based on the same principle as hash chain values, however they follow the structure of tree authentication where the values to be authenticated are placed as the leaf nodes of a tree. In order to authenticate a leaf node, the sender node discloses the values along the branch from the root of the node. Each internal node of the tree is derived from its two child nodes. The root

value is used to commit to the entire key and successive hashing of the values along the branch of the tree, the final result can be verified against the root value.

Hash tree Chains In order to prevent the malicious modification of the hop count by an adversary to show the distance from a sink node to be less than actual, we can use Hash tree chains, originally proposed in [34]. One way hash chains provide an efficient mechanism [34] for securing the hop count. However an adversary might forward the message without incrementing the hop count. Hash tree chains prevent the same distance fraud. The assumptions while using the hash tree chains are that there is an underlying mechanism for exchanging keys in the network. We have already proposed the use of Digital envelope scheme [5] in order to establish the secret keys for encryption.

The second assumption is that the protocol should be secure against replay attacks. We have shown in Section 7.1 that the protocol is secure against message replay attacks. Hash tree chains use a special one way chain, where each element of the chain encodes the node ID, thus forcing a node to increase the distance metric if it wants to encode its own ID. Each value in the one way chain contains a collection of values, one or more of which are used to authenticate any particular node. These values are authenticated using a Merkle tree and the root of the Merkle tree is used to generate the collection of values in the next step. As described in [34], a hash tree is constructed between two consecutive values v_{i-1} and v_i as follows. We use the value v_{i-1} as the input to the hash function and obtain a set of values b_0, b_1...b_n, where b_j= $H[v_I \| j]$. A hash tree is then constructed as done for the Merkle trees. The root of the tree becomes the previous value of the one way hash chain $v_{i-1} = b_0 n$. The node releases values which help to compute the one way hash tree chain forward, to verify the authenticity of the values release and use the last value obtained to sign their own id when forwarding the message, thus automatically incrementing the distance metric. In a small network, each b_j can correspond to single node, thus an attacker can not manipulate a value received from its neighbors, it must follow the hash tree chain to the next step.

To scale the solution for larger networks, a γ tuple can be used to authenticate each node. The probability of an attacker learning the values of the tuple from its neighbors and malicious modifying the message may be small [34]. A detailed analysis of hash tree chains has been given in [34]. The hash tree chain has been proved to be secure against the impersonation attack and same distance fraud. In our case, hash tree chains can secure the protocol against the attack by a single adversary or multiple non colluding single adversaries in the network, however additional mechanisms need to be employed to secure the protocol against an attacker chain as mentioned in Section 5.4.3.

Per hop hashing The hash chains are used for per hop hashing in the following way. The sender uses a random seed to generate a hash value, initializing the hash field as described above. It also specifies the hash function identifier and computes the Top Hash value. The following values are included in the message, as discussed in [17]:

- Hash value h ,which is the value obtained by applying the hash function on the previous value.
- Hash function H, $H(\cdot)$ which is a hash function identifier.
- Hop Limit, a fixed value n which determines the maximum number of hops a packet can do in a network.
- Top hash, $H^n(s)$ where s is the random seed chosen.
- Hop Count, which specifies the number of hops from the origin, that is, the sink.
- Node.Hop Count, number of hops between origin, the sink in our case and the particular node.

An intermediate node i, stores the incoming hash value $H^{i-1}(s)$ which can be used later during verification. The intermediate node increments the hop count by 1 and hashes the previously received value. The node verifies whether $H^{Hoplimit-Hopcount}(h)$= *Top hash* is true. The authors in [17] have proposed to use the field Node.Hop count to indicate the number of hops between the origin and originating node. This value can be included in a separate block in the message which is signed by the intermediate node making the modification. This value is set by the intermediate node and can be included in an intermediate node's signature and guarding against the dissemination of wrong unauthorized routing information [17].

Eliminate hop count The hop count can be eliminated and another metric can be used to determine the distance from the sink nodes, for example, packet propagation delay has been used as a metric in [16]. The originator of the packet puts a timestamp in the message and a node can compute the time a message took in reaching it. The faster routes can be used to update the routing table. However the employment of this metric requires time synchronization. A number of attacks on time synchronization protocols have been studied in [31]. A substantial amount of research has been done on secure protocols for achieving time synchronization [50].

Authentication

The control packets used for field set up can be authenticated using any one of the following techniques: μTESLA , TIK (TESLA with Instant Key disclosure), digital signature scheme or MACs. Each control packet should be authenticated by the originator. These can be used for broadcast authentication as well. The modification in the mutable field in the message can be authenticated by the sender node using these techniques as well. An intermediate node before updating should be able to verify the packet authenticity. A brief description of these schemes is given below:

μTESLA μTESLA [40] is based on TESLA where TESLA(for Timed Efficient Stream Loss-tolerant Authentication), uses only symmetric cryptographic primitives such as pseudorandom functions (PRFs) and message authentication codes (MACs), and is based on timed release of keys by the sender [37]. The scheme is based on the idea of introducing asymmetry in the symmetric cryptographic operations by means of delayed disclosure of symmetric keys. TESLA and subsequently μTESLA assume that the nodes have loosely synchronized clocks. More specifically, the scheme is based on the following idea: The sender node commits to a random key without revealing the key. The sender node then computes a MAC with the random key and transmits it along with the next packet P_j. In the next packet P_{j+1}, the sender node reveals the key and the intermediate nodes can then verify the commitment as well as the MAC. An important prerequisite in this case is that the decommitment should not have have been transmitted before packet P_j. To start this scheme, the sender uses a regular signature scheme to sign the initial commitment. All subsequent packets are authenticated through chaining [37].

TESLA is however not designed for resource constrained networks, for example, sensor networks. μTESLA modifies certain features of TESLA which reduce the overhead and enable the protocol to be used for resource constrained environment as well [40]. The sender node generates a key chain of values by using a cryptographic hash function as has been described in Section 6.2.1. Each value of the key chain is associated with one particular time slot. The keys are released in the reverse order such that node can compute the next key given the previously received keys. All packets sent within one time slot are authenticated by the same key. Initially the sender node sends a commitment, say K_0. In the first time slot, the sender node transmits the data packets with a MAC generated using the secret key K_1. None of the packets can be authenticated yet. In the next interval, the transmitted packets have a MAC computed by using key K_2 where K_1 is generated by applying the hash function on K_2, that is, K_1= $H(K_2)$. The key K_1 is disclosed two intervals after the MAC generated from the key was used. After obtaining the key, the intermediate node can verify the authenticity by verifying the MAC value. Key disclosure is independent from the packets broadcast, and is tied to time

intervals. In μTESLA, the sender broadcasts the current key periodically in a special packet. A detailed description and analysis of the scheme has been given in [40].

TIK TIK (TESLA with Instant Key disclosure) uses temporal packet leashes [33], that is, temporal information added to a packet to restrict the maximum distance traversed by the packet or the time since its original transmission. All nodes must have tightly synchronized clocks. Key authentication is accomplished using hash trees, which are an optimization of the one-way hash chains discussed in Section 6.2.1. TIK is an extension of TESLA; the key is disclosed in the same packet . However TIK requires tighter time synchronization than TESLA. In TIK, the sender computes a chain of elements using a hash function as described in Section 6.2.1. Various schemes for the same have been proposed in [33]. Hash functions are faster than encryption or digital signature schemes [34]. The sender chooses a key expiration interval for every key, i.e. every element of the hash chain and constructs a Merkle tree [51] to commit to the chain. The sender transmits the packet with four parts, a MAC computed using the key, The message payload, the tree authentication values required to compute the key and the key used to generate the MAC. Based on the tight clock synchronization, when a receiver receives the MAC, it can verify whether the key has already been released or not. If the sender has not yet started sending the key, then the receiver verifies the authenticity of the key using the tree authentication values and then uses the key to check the authenticity of the MAC. A detailed analysis of the protocol can be found in [33]. However the protocol might not be suitable for a resource constrained environment like sensor networks [33]. For sensor networks, other schemes like μTESLA can be implemented. Another optimization which the authors have proposed in [33] is that if time is too less for timestamps and digital signatures as TIK is based on tight time synchronization, more efficient authentication schemes like Schnorr signatures [43] or symmetric key MACs can be used or the packet length can be increased.

Digital signatures Digital signature schemes based on RSA [41], provide means for authentication and non repudiation. However the overhead of computing a signature and its verification may be more than some schemes like TESLA or TIK, which are based on symmetric primitives [33]. But the digital signature scheme does not require tight clock synchronization and can be used in a network where the PKI (Public Key Infrastructure) is implemented.

Other signature schemes

- **EMSS: Efficient Multi-chained Stream Signature** EMSS [37] achieves the security objective of non repudiation for a sequence of packets along with sender authentication. No buffering of packets is required at the sender. The security scheme operates on the following principle: a packet P_j includes the hash value of the previous packet P_{j-1}. At the end of the stream of packets, the sender includes a signature packet which contains the hash value of the final packet along with its signature. The authors in [37] have proposed the use of multiple hashes of previous packets in order to provide robustness against packet loss. However the receiver can verify the signature of a packet only after it has received the next signature, hence there is a delay involved in the process of packet verification. Some efficient schemes using EMSS have been suggested which provide signature verification with the RSA scheme [41] and offer low overhead with high verification probability in [37]. Other related schemes have also been discussed in [37].
- **Aggregate signatures** Aggregate signatures allow to join n signatures on n distinct messages from n distinct users into one. This approach has been used in [17] and there are two aggregate signature schemes based on Short Signature scheme which supports General Aggregation Schemes (GAS) and Multisignature schemes which support Sequential Aggregation (SAS). A multisignature scheme can be used when immutable fields are signed by the nodes because of their potential efficient batch verification properties [17].
- **Schnorr signatures** An efficient signature scheme has been suggested in [43] which is based on the problem of discrete logarithms and replaces the verifier's challenge in the

authentication schemes by a hash chain. The detailed protocol description for signature generation and signature verification has been given in [43].

MAC shared keys Message Authentication Codes can be used independent of the approach adopted by the above schemes as well.

Polynomial Verification techniques A novel lightweight technique for authentication of messages has been proposed in [49]. The scheme uses verification of polynomial based messages in order to authenticate the sender. The message authentication and computation overhead resulting from this technique is less as compared to public key cryptography and unlike TESLA and TIK, this approach does not require tight clock synchronization. As the scheme is lightweight and involves less communication overhead, it has been suggested for sensor networks. A detailed description of the scheme can be found in [49].

Additional measures In addition to these there are certain additional parameters or solutions which we propose to include in the protocol in order to secure the protocol against certain specific attacks. These are:

Against misbehaving nodes

In order to secure the protocol against misbehaving nodes, a mobile intrusion detection system [44] can be installed in the network. The neighboring nodes of an adversary can detect a misbehaving node and report the event. Various reputation based systems have been proposed in literature. A survey of some of the proposed systems can be found in [53].

Another solution is to maintain a set of metrics which are indicative of how the node has behaved in the past. On the basis of these ratings, a misbehaving node can be detected. However this measure is liable to various secondary attacks, for example, a node may raise false alerts and lead to an honest node being falsely suspected. Also the reporting nodes would need to be validated. Another solution proposed in literature is that of using fictitious currency in order to provide incentive to the nodes to behave in accordance with the protocol. However this solution assumes a Public key infrastructure and establishment of an online Certification Authority. Some of these solutions have been reviewed in [22].

Another solution may be to use trust and privilege levels in order to protect the message integrity. Trust levels have been used in ARMA [18] and as part of the SAR protocol described in [59].

Watchdog/Pathrater approach has been proposed for DSR [13] in [46]. The local network traffic is monitored for anomalies and based on the link performance and reliability according to these parameters, the pathrater evaluates the routes. The routes are chosen according to their ratings. There are some drawbacks to this approach as a bad node can cause an honest node to appear malicious or the effect of a bad node can be averaged out by several honest nodes in a long route.

Another approach as used in [17] states that selfish attacks are prevented with the mandatory inclusion of the previous nodes signatures in the routing element. *The more signatures the protocol forbids to remove, the bigger the coalition of nodes required for selfish attack [17].*

Against Neighbor discovery attacks

Packet Leashes Packet leashes [33] can be used to secure the protocol against neighbor discovery attacks. However packet leashes require precise time synchronization or GPS location service and loose time synchronization. Packet leashes can be geographical or

temporal. Geographical leashes require that each node is aware of its own location and assume loose time synchronization. The sender includes its own location and the time at which the packet was sent. By comparing these values with the time at which the packet was received and its own location, a receiver node can compute an upper bound on the distance between itself and the sender. A timestamp or digital signature can be used to authenticate the location claim. A node can further verify if a wormhole is present or not by checking the radius around the sender and itself. The TIK protocol used to implement the temporal leashes has been described in Section 6.2.1. This approach is used to detect wormholes. However if timestamps are used, in the case that a node gets compromised, an adversary can manipulate the clock and the system is susceptible to replay attacks.

A number of other techniques, for instance the use of directional antennas to detect the transmission of messages and statistical wormhole detection techniques have also been proposed in literature. A detailed analysis of these can be found in [5].

Proof of neighbor relationship ADVSIG [25] used to secure OLSR [12] states that if node A declares a symmetric link with node B, it should include a proof signed by B obtained from previous HELLO messages sent by B. This approach counters against link spoofing attacks and secure against unidirectional links being used as bidirectional.

Another approach as proposed in SNEP (Secure Neighbor Establishment Procedure) [23] is that each node broadcasts the following information to its neighbor:

- List of nodes from which it received a hello packet
- Hash value of combination of the recently received hello messages from every neighbor and the node's ID, for instance, node A would broadcast the list of neighbors from which it has received hello packets and also send the hash value, $h(h(m_B), A)$ where A is node A's ID and m_B is the hello message received from B.

The hash value serves as a proof that the link really exists. A node validates another node as its neighbor after receiving the hash value as described.

Secure Localization Techniques A number of techniques have been proposed in literature which can be used to detect the wormholes and sybil attack as well. Some techniques, for instance, distance bounding and distance estimation techniques have been discussed in [57]. Secure localization mechanisms can be used to prevent attacks related to location. Secure localization is an important requirement in sensor networks. Other distance bounding techniques include Mutual Authentication Distance bounding (MAD protocol) discussed in [5].

Against impersonation

Cryptographically Generated Addresses Cryptographically generated addresses [5] protect against some neighbor discovery attacks.These addresses also prevents a node from stealing already chosen addresses. A detailed description has been given in [35].

Radio fingerprinting Radio Frequency Fingerprinting (RFF) has been used to uniquely identify a given transceiver, based on its transceiver print. In [38], the authors propose to associate a MAC address with the corresponding transceiver profile. The use of this technique along with an Intrusion Detection System (IDeS) can help to prevent several attacks, for example, impersonation of nodes, node replication attack and so forth. This technique yields high success rates in terms of classification of transceiver prints. A detailed analysis of the scheme has been presented in [38].

Against Black hole/gray hole attacks

A Watchdog/Pathrater [46] approach can also be used to detect a black hole. Other techniques proposed for detection of misbehaving nodes can also be implemented for detecting a black hole.

Sybil Defense mechanisms

Some of the defenses against Sybil attack are:

- Key predistribution process: The scheme has been discussed in detail in [5].
- Verify location claims: Secure localization techniques [57] can be used to verify the location claims and accordingly thwart the adversary's claims of various entities as discussed in Section 6.2.1.
- Radio Fingerprinting [38], which has been briefly discussed in Section 6.2.1, can be used to detect a sybil attack with the help of an installed Intrusion Detection System (IDeS).

Against jamming

The problem of jamming at the physical layer is beyond our scope. Some solutions which have been proposed in the past include the use of frequency hopping, spread spectrum techniques. A number of countermeasures have been suggested by Wood and Stankovic in [47] for link jamming, for example, the use of error correcting codes for collision attack, using rate limitation for exhaustion attack and using small frames against unfairness attacks. However the problem of link layer jamming is orthogonal to our goal and will not be discussed further.

Against Node Replication

Node replication detection schemes Two distributed schemes have been proposed in [39] for node replication, *Random Multicast* and *Line Selected Multicast*. The first approach, **Random Multicast** involves distributing location claims to randomly chosen witness nodes in the network. A node announces its location claim to its neighbors who forward the message to a randomly chosen set of witness nodes. If the adversary replicates a node, accordingly two different sets of witness nodes would be chosen. The birthday paradox is used to predict the chances of a common witness node receiving the replicated node claims. According to the birthday paradox, in a set of n nodes, if each location claim is sent to $\sqrt{n}$ nodes, minimum one collision is predicted with a high probability,that is, at least one witness will receive a pair of conflicting location claims. The witness node receiving similar claims can then inform the Intrusion Detection System, if installed in the system or flood the network with a message to revoke the replicated node. However the approach presumes that each node is aware of its own location. For the purpose of authenticating the location claim, any of the authentication schemes suggested in Section 6.2.1 can be used. Various enhancements have also been proposed in [39] to lower the communication overhead and storage space required.

The second scheme, **Line Selected Multicast** uses the routing topology in the network in order to select the witness nodes and uses geometric probability for detecting replicated nodes. In order to lower the communication costs, the authors propose a different mechanism for selecting the witness nodes in the network. 'When a node's neighbors send out the evidence of its location claim to the r witnesses, each of the nodes along the route stores a copy of the location claim as well. If these intermediate nodes also store the location claim, then we have effectively drawn a line across the network. If a conflicting location claim ever crosses the line, then the node at the intersection will detect the conflict and initiate a revocation broadcast' [39]. As an example, consider a node i which sends its location claims to a certain node in the network via a route of other nodes. The node receiving the location claim verifies the location claim and if there is no match with the previously stored location claims, it forwards the message to the next destination. As shown in [39], if any of the nodes discover a conflicting claim, the pair of location claims are flooded in the network and revocation of the

nodes in question can be done. If the collision happens to occur at a replica, it still does not preclude another collision from occurring elsewhere in the network. The decisions involved in this approach are local and are non deterministic, hence an adversary cannot tamper with the mechanism unless it has a very strong adversarial control over the network.

Even though the schemes have been proposed for sensor networks, these can be implemented in other wireless ad hoc networks as well. These approaches can be integrated with the Cluster approach suggested in Section 6.2.2 in order to provide means for potential verification as well. The approach has been discussed in the next section.

Centralized approach Other alternatives include a centralized approach wherein each node sends a list of all its neighbors' claimed locations to the central authority which detects the node replication attack. However this approach assumes the presence of a central infrastructure and is also susceptible to attacks as there is a single point of trust in the network [39]. Another approach can be to bind Node IDs to their MAC addresses.

Local detection can also be done where the neighboring nodes detect a replicated node by means of voting. However these means fail to secure the protocol against distributed attacks of this nature in disjoint neighborhoods [39].

Radio Fingerprinting [38], which has been briefly discussed in Section 6.2.1, can be used to detect a node replication attack with the help of an installed Intrusion Detection System (IDeS).

Detecting tunnels

Tunnels and wormholes are similar to some extent however tunneling is a network layer attack where one of the adversary encapsulates routing packets into normal data packets and sends them to another colluding node. The presence of a tunnel in the network can result in resource consumption of nodes which lie on the route between the adversarial nodes and storing of incorrect routing information. A number of attacks can be mounted where an adversary claims a false neighbor relationship. To detect a tunnel, certain centralized wormhole techniques like statistical detection [5] can be used. Secure localization techniques where each node is aware of its own position and the location claims can be verified can also be used to counter the problem.

ARAN [16] uses packet propagation delay as a metric instead of hop count. Hence an approach to detect tunnels could be based on the same principle because there would be more delay on a virtual link than on an actual route [5]. A second approach suggested in [5] is to measure the round trip time to its three hop neighbors explicitly and from these values, a node can detect whether a neighbor is reachable via a tunnel or not.

Protection against Denial of Service attacks and Flooding attacks

As discussed in [9], the approach can be:

- Account for all resources consumed by every user (or flow)
- Detect when resource consumption by given user exceed those allowed by system policy
- Reclaim consumed resources

Rate limiting techniques [24] can also be enforced in order to prevent injection of packets into the network by an adversarial node with the aim of resource consumption.

6.2.2 Gradient Establishment

The aim is to exchange potential values with one hop neighbors. As discussed in the system model, after a node has received minimum one message from each of its neighbors, it calculates its value according to the potential function. After calculating the potential value, the node sends a message indicating its potential value to its one hop neighbors. All the fields in the message are immutable and are signed by using any of the following schemes: μ tesla, TIK, digital signature scheme or MACs. The fields are {Potential, SenderID, DestinationID, TimeSlot, CTR}, as shown in Figure 6.3.

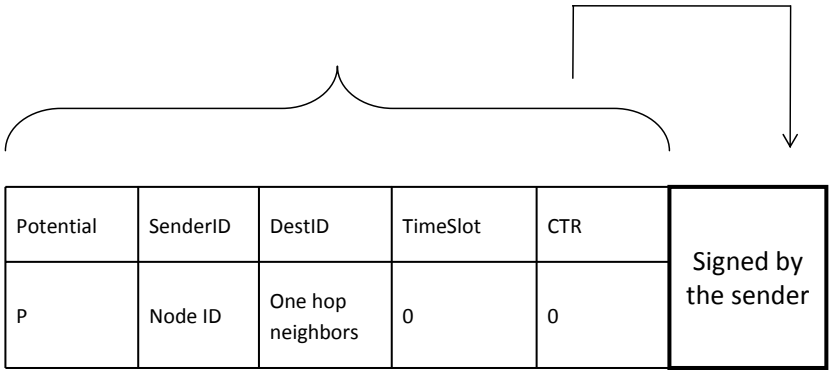

Figure 6.3: Gradient Establishment

Verifying potential All the fields indicate the same parameters as in case of potential field establishment; however, the potential value here is set to the value of the sender node's potential and since the message is only forwarded to the one hop neighbors, the hop count is not required to be changed. In order for the potential to be verified, we propose two approaches, either of which can be used depending on the application scenario:

Local potential verification approach

A node also sends the messages it used in order to calculate the potential value. The one hop neighbors of the sender node verify the signature and check the potential value computation. If the verifications are successful, the neighbors add the entry to their tables or update the entry in case an entry already exists, else the packet is discarded. The nodes choose the neighbor with the highest potential for a destination group for the next hop. In case the verification fails the honest node drops the packet. If there is an Intrusion Detection System [44] operating in the network, then appropriate steps can be taken to isolate the adversary and transmit the message to other nodes in the network. Intrusion detection systems have been studied in [44] and these systems describe the approach followed after an adversarial node is identified or malicious behavior is detected. This approach can be used to detect a single adversary or multiple non colluding single adversaries in the network. However for detection of a chain of adversarial nodes, additional mechanisms would be needed as described in Section 6.3. Thus this approach can be used for verification of potential by one hop neighbors. The approach has been briefly described in Figures 6.4 and 6.5.

We consider a network topology as shown in Figure 6.4 in which the potential field messages are flooded across the network. All the nodes use these field messages to compute their potential value and send these values to their neighbors for determination of the steepest gradient. The local verification mechanism has been described in Figure 6.5. We consider the verifications performed by Node *N*. Each neighbor of Node *N* sends a message indicating its potential value and also the messages it used to calculate that potential value. Node *N* verifies the potential value and accordingly updates its routing tables. Note that if the adversary A provides a wrong potential value, it would be detected on the basis of potential field messages.

Cluster approach

The second approach involves the division of the network into verification clusters randomly. The network is divided into clusters and each cluster has a cluster head, which verifies the potential value for the other nodes in the cluster. The motivation for this approach is that in order to verify a neighboring node's potential, a node might require a large amount of information about the network. A number of algorithms have been proposed in literature for cluster formation and cluster head election. Some of these have been reviewed in [44]. In order to verify the potentials, a cluster head is formed, which has the information about the nodes in its cluster. All the nodes send the message containing their potential value to the cluster head. The nodes which compute their potential value using other nodes which are not members of the same cluster, also send the messages received from these nodes to the cluster head for potential verification. However the overhead in that case would be larger than the first approach in case of verification of potential by one hop neighbors. In case the network is highly dynamic, the first approach can be preferred for securing the protocol against single adversarial nodes or multiple non colluding adversaries in the network. But the cluster approach would help to mitigate the attacks from the more powerful adversaries and can be used to detect a chain of attackers as well. A cluster based intrusion detection system has been described in [44]. The cluster formation process is randomized, hence the attacker would need to know beforehand about clusters formed in order to disrupt or influence the process. The approach has been briefly described in Figures 6.4 and 6.6.

We again consider the network topology as shown in Figure 6.4 in which the potential field messages are flooded across the network. All the nodes use these field messages to compute their potential value and send these values to their neighbors for determination of the steepest gradient. The Cluster approach has been described in Figure 6.6. We consider the cluster as shown, formed by one hop neighbors of *N* which is the cluster head. The potential verifications are performed by Node *N*. Each node in the cluster of Node *N* sends a message indicating its potential value and also the messages it used to calculate that potential value. Nodes A, E, G use the messages sent by nodes within the cluster for potential calculation, hence these nodes do not need to send the potential field messages received from nodes in the same cluster to the cluster head. Only Node *N* verifies the potential values. Note that if the adversary A provides a wrong potential value, it would be detected by the cluster head on the basis of potential field messages.

A list of all the above solutions has been presented in Table 6.1. The list is not comprehensive.

6.2.3 Packet Forwarding

The motive is to forward data packets along the steepest gradient to the neighbor with the highest potential. The fields in the data packet are: {SenderID, DestinationID, TimeSlot, CTR }, as shown in Figure 6.7, which are signed by the sender node using any one of the following techniques: μ tesla, TIK, digital signature scheme or MACs. The authentication schemes have been discussed in Section 6.2.1.

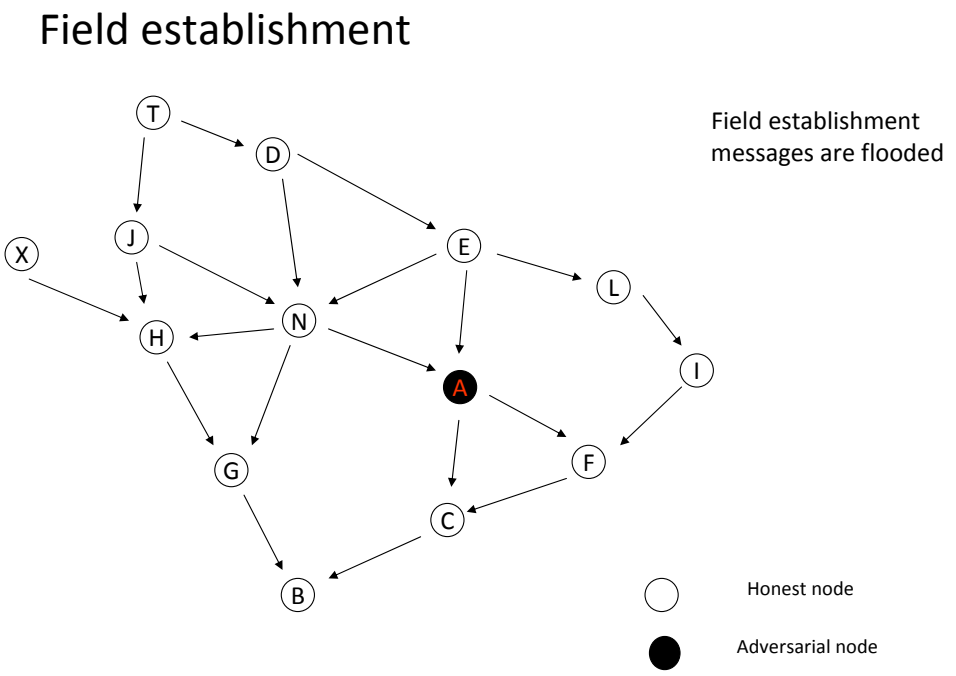

Figure 6.4: Field Establishment: local verification and cluster approach

We consider a network topology as shown above in which the potential field messages are flooded across the network. All the nodes use these field messages to compute their potential value and send these values to their neighbors for determination of the steepest gradient. The two approaches for potential verification for this topology have been shown in Figure 6.5 and Figure 6.6.

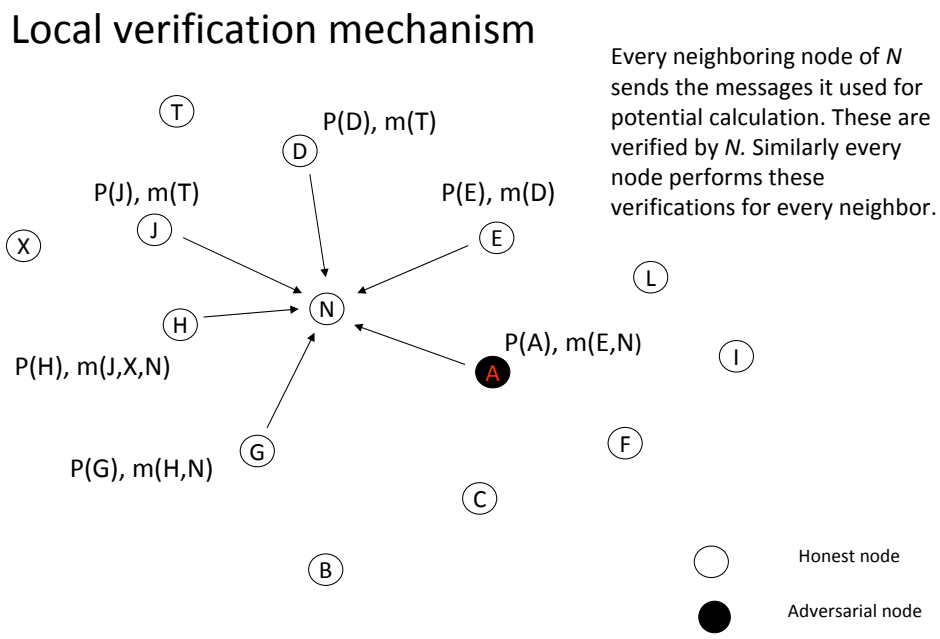

Figure 6.5: Local verification approach

The local verification mechanism has been shown for the topology described in Figure 6.4. We consider the verifications performed by Node *N*. Each neighbor of Node *N* sends a message indicating its potential value and also the messages it used to calculate that potential value. Node *N* verifies the potential value and accordingly updates its routing tables. Note that if the adversary A provides a wrong potential value, it would be detected on the basis of potential field messages.

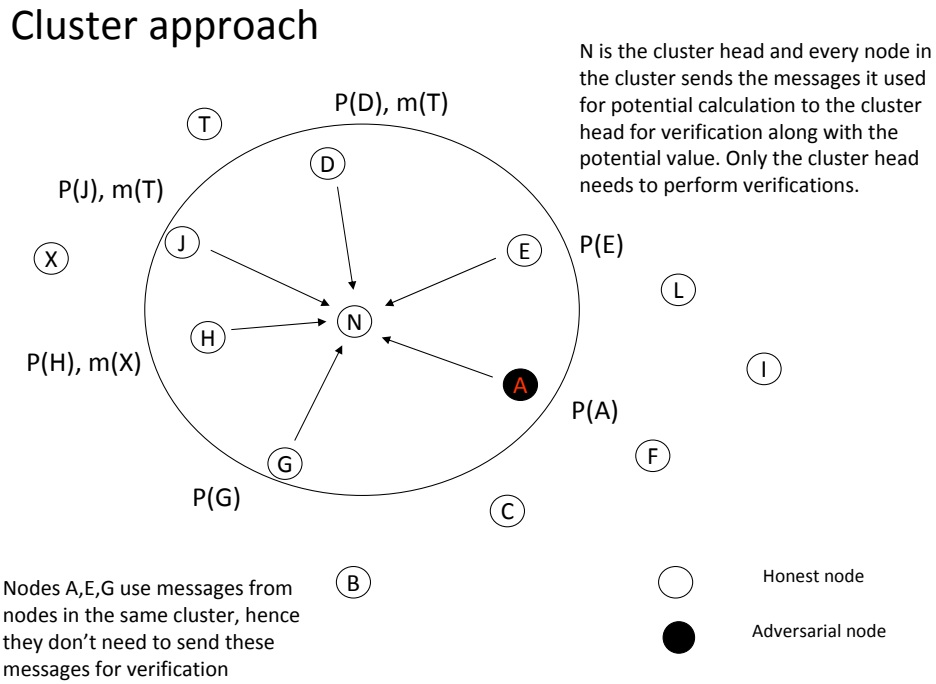

Figure 6.6: Cluster approach

The Cluster approach for potential verification has been shown for the topology described in Figure 6.4. We consider the cluster as shown, formed by one hop neighbors of *N* which is the cluster head. The potential verifications are performed by Node *N*. Each node in the cluster of Node *N* sends a message indicating its potential value and also the messages it used to calculate that potential value. Nodes A, E, G use the messages sent by nodes within the cluster for potential calculation, hence these nodes do not need to send the potential field messages received from nodes in the same cluster to the cluster head. Only Node *N* verifies the potential values. Note that if the adversary A provides a wrong potential value, it would be detected by the cluster head on the basis of potential field messages.

Solutions Available		
Securing mutable Information	Hash chains Hash tree chains Eliminate hop count	Tree authentication Per hop hashing
Authentication	μTESLA Digital signatures Aggregate signatures MACs	TIK EMSS Schnorr signatures Polynomial verification
Against misbehaving nodes	Watchdog/Pathrater Use incentives : fictitious currency	Maintain reputation values Include intermediate node signatures (against selfish attacks)
Secure Neighbor discovery	Packet leashes Distance bounding protocols Securing location Include hash of hello message and node ID as proof	Centralized wormhole detection techniques Plausibility checks Show signed hello messages as proof
Against Impersonation	Cryptographically generated addresses Polynomial verification techniques	Radio fingerprinting
Against Sybil	Verify location claims Key predistribution	Radio fingerprinting
Against node replication	Random Multicast Centralized approach	Line selected Multicast Radio fingerprinting
Detecting tunnels	Secure localization techniques Measure RTT to 3 hop neighbors	Use propagation delay as a metric Plausibility checks
Against insertion of messages and flooding	Rate limitation techniques	
Potential verification	Local verification by one hop neighbors	Cluster approach

Table 6.1: Summary of Solutions

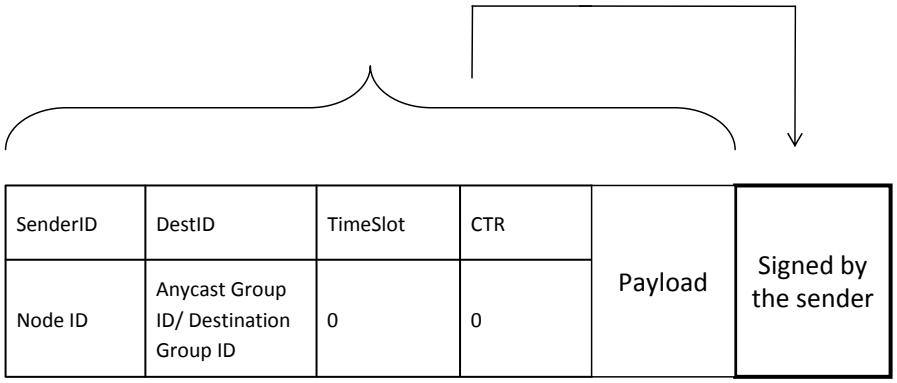

Figure 6.7: Packet Forwarding

All the field are immutable. The intermediate nodes verify the signature and the origin; if the origin is verified, the intermediate node forwards the message along the steepest gradient to its neighbor with the highest potential.

Next, we present five approaches that combine some of the above explained security mechanisms.

6.3 Secure Approaches for field setup

We propose five distict approaches which can be used to secure the protocol in different application scenarios. In order to thwart the attempts of the adversary, we need to find a way of securing the change in the hop count as well as a secure signature scheme for ensuring the authenticity and origin authentication of the message. We need to ensure that the messages during the potential verification (in case we use the first approach for gradient establishment phase of sending messages used for potential computation) are not misused by the neighbor and find means of reducing the message overhead in the secure protocol, that is, an efficient flooding mechanism. The analysis of these approaches has been given in the next section.

There are two basic flooding mechanisms which can be used in context of the field theoretic approach. These are: broadcast mechanism [1] and sending messages to local one hop neighbors for establishing the potential field [3]. We have proposed two different mechanisms for verification of potential, potential verification by verification of messages and the cluster approach. **Approach 1** uses the local flooding mechanism for potential field set up and the former approach for verification of the potential. **Approach 2** uses the broadcast mechanism for establishing the potential field and the former approach for verification of potential values. **Approach 3** uses the local flooding mechanism for establishing the potential field and the cluster approach for potential verification. **Approach 4** uses the broadcast mechanism for establishing the potential field and the cluster approach for verification of the potential values. **Approach 5** uses the broadcast mechanism for establishing the potential field and the potential of a node is verified by tracing the route of a packet from the sink to that particular node.
Any of the authentication schemes discussed in Section 6.2.1 can be used in order to authenticate the packets in all the approaches, however TIK [33] can be used for serving dual purposes in the approaches where a cluster head is not involved, that is, Approaches 1, 2 and

5. This is because it would prevent the need for employing additional mechanisms for detecting wormholes. However TIK also requires tight clock synchronization.

In view of the application where the approach is used, a suitable authentication mechanism can be chosen. Similarly, any of the countermeasures suggested in Section 6.2.1 can be used to prevent the malicious modification of the hop count in the approaches where the mutable field is present, i.e., Approaches 2 and 4. These mechanisms have been suggested to secure the FBR specific field set up attacks. Additional mechanisms that can be used to counter the general attacks have been suggested as well.

A number of countermeasures have been discussed in Section 6.2.1 in order to secure the protocol against various attacks for instance sybil attack and node replication attack. Any one of these counter measures can be implemented.

Node Isolation We have seen in Section 5.4.1 that in case of a chain attacker, node isolation can occur. The proposed technique to alleviate the impact of the attacker in such a case would be to employ redundancy in the network, that is, to deploy more number of nodes in order to prevent the case of node isolation where verifications are not possible. For all the approaches, our main attacks of interest are the ones in which the working of the routing protocol is disrupted when legitimate routes are possible from a node.

False updates Since the attacker is an insider, it has access to the cryptographic keys for minimum one compromised node. Hence it can generate false updates as well. The impact of this attack can be mitigated till some extent by the verifications performed on TimeSlot and CTR field. Also, in either of the two approaches being used for verification of potential, a node can not indicate a false potential value, as it would be subject to verification. Hence it would not be able to generate false potential updates.

6.3.1 Approach 1

This approach uses the mechanism based on use of local potential field messages to one hop neighbors for field establishment, proposed in [3] in order to establish the potential field and the steepest gradient. In order to verify the potential, a node presents the messages it used to compute its potential.

Field set up In this secure approach, for the purpose of field setup, we use the approach proposed in [3]. The hop count is not required in this approach as each node forwards the message to its hop neighbors only. The field is established according to the algorithm described in [3], that is, each node initially sets its potential to 0. After the gateway nodes transmit their potential to their one hop neighbors, these nodes calculate their potential accordingly and forward the message containing their potential value to their one hop neighbors In this way the potential field is established. The nodes arrange their neighbors in descending order of their potential values and use the algorithm in [3] to calculate their own potential value. Any of the mechanisms proposed in Section 6.2.1 can be used to authenticate these field messages. However to avoid the count to infinity problem which might arise due to the use of local messages for establishing the field, a technique like poison reverse [3] might be required. The message would only contain the fields <Potential, SenderID, DestID, TimeSlot, CTR> and all of these are immutable fields. This approach provides the advantage of combining the first two phases as described in Section 3.2.

Potential verification In order to verify the potential values, we propose that each node sends the messages it used to compute its potential along with the message containing its potential. This approach assumes a Public Key Infrastructure (PKI) as each node should be able to verify the messages signed by other nodes. Rate limiting techniques [24] are used to limit the injection of false updates in the network.

This approach can successfully detect a single adversary node or multiple non colluding single adversarial nodes in the network. However additional mechanisms need to be employed for detection of a chain of adversarial nodes. Barring the case of an isolated node, at least two neighbors in a chain would have a common neighbor. The potential verification carried out by the common node can detect the adversarial chain nodes. Using any of the mechanisms described in Section 6.2.1 prevents the hop count from being decreased. So the adversary can not effectively create a sinkhole.

If the adversarial nodes forward a message without incrementing the distance metric, these would have the same effect on the routing as a single adversary. The adversarial nodes can lead to resource consumption however, by incrementing the hop count and making nodes appear farther from the sink nodes. Alternate routes can be used in that case, if possible. Additional plausibility checks [24] for the velocity, time and distance traveled by the packet are carried out by individual nodes to determine whether the packet has been been transmitted via a tunnel or by an out of band connection. The authentication scheme TIK, described in Section 6.2.1 can be used with this approach, it would serve the dual purpose of authentication as well as detecting wormholes by means of implementing packet leashes.

6.3.2 Approach 2

This approach utilizes the broadcast mechanism for field setup and uses the local verification approach for verification of potentials. Any of the mechanisms suggested in Section 6.2.1 can be used to prevent the malicious modification of the hop count.

Field set up In this approach, we propose to use the broadcast mechanism proposed in [1] for the purpose of broadcasting the potential values of the sinks. As discussed in Section 3, the sinks broadcast their potential as infinity. The message format is as shown in Figure 6.2. The intermediate nodes modify the hop count and sign the message. Each node which receives the message, modifies the hop count, removes the signature of the previous nodes and signs the entire message. Thus the message is a concatenation of two parts: first, the static information signed by the sink and second, the modified immutable field as well the first part, signed by the intermediate node. The nodes exchange messages containing their potential values with their neighboring nodes in order to establish the steepest gradient.

Potential verification The verification of potential messages is done via exchange of messages as described in Approach 1 by means of sending the messages used to compute the potential along with the message containing the node's potential.

6.3.3 Approach 3

This approach uses the mechanism proposed in [3] in order to establish the field, and the cluster based approach in order to verify the potential.

Field set up The field set up procedure is the same as proposed in Approach 1 which uses transmission of messages locally (to one hop neighbors) in order to establish the potential field. However the approach used for potential verification is different.

Potential verification In order to verify the potential, we propose to use the cluster based approach described in Section 6.2.2. A number of algorithms have been proposed in literature which discuss the formation of clusters in a randomized way. One of these has been discussed in [44], where clusters are formed and the leader is elected in a randomized manner. The attacker has lesser chances of disrupting the routing if the approach used is randomized. After the field is established as described above, all nodes send a message containing their potential value to the cluster head. The cluster head checks for anomalies in the potential messages. If a node gives higher potential than expected or an adversary tries to create a sinkhole, the anomaly would be detected and the cluster head can broadcast a message about

the adversarial node or in case of an IDeS installed in the system, the cluster head can send a message to the central authority. It is possible that a node uses the messages from its neighbors which are not in the same cluster as it is. In order to prevent the higher potential from being mistaken as an anomaly, the node includes the messages it has received from its neighbors which are not in its cluster. These messages are also sent to the cluster head which makes the verifications. The cluster head has information about the nodes in its cluster.

In case an adversarial node is elected as a cluster head, it can create a sinkhole and disrupt the routing by false claims against honest nodes and malicious modification. However a leader is elected after a certain stipulated period and the method used is fair and random. Thus, when the next leader is elected, the adversarial control over the cluster would also be lessened. In addition, a cluster head can periodically monitor the nodes in its cluster and can verify their claims.

The mechanism proposed above where each node which has neighbors in another cluster, forwards the messages from its neighboring nodes to the cluster head, can secure the protocol against a single adversary or multiple non colluding single adversaries in the network. In order to verify the potential of a node when a chain of adversarial nodes is present in the network, a cluster head has to periodically exchange information about the node potential with the cluster head of a neighboring group. These messages can be authenticated and protected from malicious modification using any of the techniques presented in Section 6.2.1.

6.3.4 Approach 4

This approach combines the broadcast mechanism for the field setup with the cluster based approach in order to verify the potentials. Any of the mechanisms suggested in Section 6.2.1 can be used to prevent the malicious modification of the hop count.

Field set up The approach used for creating the potential field is the same as the broadcast method suggested in Approach 2. The message format is as shown in Figure 6.2.

Potential Verification The cluster based approach, described in detail in Approach 3 is used for verification of potentials.

6.3.5 Approach 5

This approach is based on tracing the route traversed by a packet in order to verify its potential. The broadcast mechanism is used in order to establish the potential field in the network and the nodes present the messages used in order to calculate their potential value for potential verification. Additionally TIK could be used for authentication as it implements packet leashes. Plausibility checks are used to detect a tunneled packet.

Field set up The sink nodes broadcast their potential to be at infinity. However all the fields in the message are immutable and these include, {Potential, SenderID, DestID, TimeSlot, CTR}. The sink authenticates the message using any of the authentication schemes described in Section 6.2.1. An intermediate node on receiving the message includes its own ID in the message and forwards the message. The next node does not remove the previous node's signature but adds its ID to the list and signs the message. The aggregated signature scheme, discussed in Section 6.2.1 is used here. The nodes determine their distance from the sink by means of the number of identifiers in the list and accordingly compute their potential.

Potential Verification A node presents the messages it used in order to compute the potential. The potential value claimed by a node can be verified by tracing the route of the message the node presents as proof as it includes the signature of the sink node and all the intermediate nodes on the route from the sink to the node performing the verifications.

Approach	Flooding mechanism	Potential verification
1	Local	Local verification by showing messages
2	Broadcast	Local verification by showing messages
3	Local	Cluster approach
4	Broadcast	Cluster approach
5	Broadcast	Tracing packet route: showing messages

Table 6.2: Secure Approaches

A summary of the security mechanisms used in secure approaches has been shown in Table 6.2.

6.4 Sensor Networks

Sensor networks have gained a lot of attention in the research community lately as a consequence of their diverse applications such as monitoring safety and security of buildings and spaces, measuring traffic flows, and tracking environmental pollutants. Sensor networks will play an essential role in the upcoming age of pervasive computing, as our personal mobile devices will interact with sensor networks in our environment [57]. However the devices used in sensor networks are relatively more resource constrained than in other wireless ad hoc networks and have some additional security requirements as well. Some of these, as discussed in [57] are:

- Authentication
- Secrecy
- Availability
- Service Integrity

Authentication For the purpose of authentication in case of resource constrained sensor networks, we can use Elliptic curve cryptography [5], which is the most compact form of public key cryptography. Other authentication schemes which have been discussed in Section 6.2.1 and can be considered for use in sensor networks are polynomial verification [49] or μTESLA [40]. Another protocol suggested for sensor networks in [40] is Secure Network Encryption Protocol (SNEP), which meets the objectives of data authentication, data confidentiality, message freshness and data integrity.

Secrecy For the purpose of confidentiality, we have proposed the use of digital envelope scheme [5] in Section 7.1. However in view of the constrained resources, other schemes can also be explored for the same. A detailed discussion of schemes for key establishment can be found in [5]. SNEP [40] can also be used to meet the objective of confidentiality.

For the purpose of potential verification, the use of cluster based approach can be considered. The approach can be adapted according to the requirements of the sensor networks and the base station can be used as a central point of trust for certain tasks,for instance, for establishing and maintaining the cluster heads. The cluster based approach has been discussed in detail in Section 6.2.2.

Use of secure localization techniques can also help to mitigate a number of attacks for instance, node replication and sybil attack, as discussed in Section 6.2.1. The schemes proposed for replication detection : random multicast and line selected multicast, as discussed in Section 6.2.1, have been proposed in the context of sensor networks and hence can be applied. A more comprehensive analysis of sensor network related issues can be found in [54]. Other related objectives like prevention of traffic analysis, resistance of a captured device, return routability , time shifting attack and so forth, have been studied in [5].

6.5 Summary

In this chapter, firstly, we presented the measures to secure the protocol against the outsider attacker. Next, for the purpose of securing the protocol against the insider attacker, we discussed the various countermeasures in general, which can be used to safeguard the field theoretic approach against specific attacks. We primarily discussed two main mechanisms for establishing the potential field, broadcast mechanism or field establishment via exchange of local messages. We proposed two main approaches for the purpose of verification of potential values: local potential verification by showing messages used in the potential calculation and the cluster approach. By combining these mechanisms, we proposed five different approaches which can be implemented in order to secure the field set up in FBR. Some approaches which can be used specifically for the sensor networks were discussed in the concluding section.

The secure approaches presented in this chapter are analysed in the next chapter. The secure protocol against the outsider attacker is also discussed. A provable secure analysis is also presented.

Chapter 7

Security Analysis

In this chapter, we analyse the secure protocol proposed in the previous chapter. Firstly, the case of outsider attacker is considered and the measures used to secure the protocol against the outsider attacker are considered. A provable security analysis for the same is also given. Next, we analyse the approaches which have been proposed to secure the field set up against the insider attacker. The results for the same are discussed in the next chapter.

7.1 Outsider Attacker

Our aim is to prove that given the assumptions, the security objectives are met even in the presence of an adversary. We prove that the measures suggested in Section 6.1 prevent the attacks by an outsider attacker.

False injection of messages in the network by an unauthorized node is prevented due to the presence of the signature scheme used (or any other authentication mechanism used). Hence a node receiving a packet which had been injected into the network by an adversary shall discard the data packet as each node carries out signature verification on the packets it receives before forwarding the packets. It also ensures the goals of non repudiation and authentication. The attacker can also not modify any of the messages due to the authentication mechanisms present. If she is not an authenticated member of the network, she would not have access to the keys for doing so.

However the authentication mechanism does not ensure protection against replay attacks. The adversary may replay an old message belonging to a previous time slot and this would lead to resource consumption and disrupt the routing as well. In order to secure the protocol against attacks of this nature, the proposed protocol contains a TimeSlot and CTR field. So any sender node is aware of the current slot in which the network is operating. For the mobile nodes which join the network at a later slot, the clocks could be set via exchange of messages with neighbors while the potential value messages are being exchanged. When an intermediate node intercepts a message, it checks the time slot field in the message. If the time slot when the message was sent is still within a certain limit the intermediate node forwards the packet, else the packet is dropped. This prevents replaying of old messages by an attacker.

The use of counter ensures that in case of an attacker intercepting data packets between two nodes and replaying the packets later, the routing process shall not be disrupted due to the ordering which is specified by the sender node in the form of counter values on each of its packets.

Even though confidentiality of messages is not a major concern considering the objectives stated in Section 5.1, in order to secure the messages against unauthenticated access, an encryption scheme can be employed. If the scheme employed is public key cryptography, then the concept of a Digital Envelope [5] can be used. Digital envelope is a hybrid encryption

technique in which the messages are encrypted with a randomly generated symmetric key, and the key is then encrypted with the public key of the intended recipient. Only the receiver node has the private key to decrypt this message and hence, only an authenticated user has access to the message. This scheme has advantages over a pure asymmetric key encryption scheme, as the latter is rarely used to encrypt large messages. Note that, if the sender and receiver already share a secret symmetric key, it can be used directly for encryption. Encryption of messages under a secure encryption scheme also prevents the unauthenticated outsider from extracting information by eavesdropping.

7.1.1 Provable Security Analysis

For the purpose of formal security analysis, we can use the ideal world model-real world model concept proposed in [19]. The concept has been proposed for source routing protocols and we extend the model for analyzing the field theoretic approach. The messages which are exchanged for calculation of potential values are the main focus for the analysis because field based approach is most prone to attacks leading to wrong potential value calculation and thus creating an incorrect routing state. We assume that all messages which lead to incorrect routing state in the form of wrong potential calculation and thus leading to local maxima shall be flagged in the ideal world model.

The definition of secure routing is the same as proposed in [19], which states that a routing protocol is secure if for any system configuration and any real world adversary, there exists a corresponding ideal world adversary such that the outputs of both the models are computationally indistinguishable.

In order to prove that the protocol is safe, we need to find an ideal world adversary corresponding to the real world adversary assumed in our model. For statistical indistinguishability, the proof technique used is the same as in [19]. In accordance with the assumptions made for the outsider attacker model, the attacker can only cause a message to be flagged if it forges the signature of the nodes. Therefore, the protocol is secure if the signature scheme is secure against chosen message attacks, the proof for which is similar to as has been proposed in [19].

An informal analysis of the protocol suggested is given below:

The main difference in field based approach as compared to topology based or source routing protocols is that there is no explicit route discovery message circulated. FBR is proactive in nature. So the attacks that are inherent to these protocols may not be effective in field based routing.

- If we only consider the outsider attacker, the attacks which can be mounted are of the nature of resource consumption mainly. However in absence of the authenticated keys and cryptographic material, the adversary can not generate authenticated messages, so the node verifying the message inserted by the adversary shall discard it.
- The second type of attack can be of the form of replay attacks in which the adversary can replay a message belonging to a different time slot. This is prevented by setting the time slot in the field TimeSlot.
- The type of attack in which attacker can replay the message in the same time slot is prevented by use of CTR field. A node numbers every packet it sends in a particular time slot. The receiver can verify the replayed message by checking its counter value.
- For countering wormholes, packet leashes can be used. However, for geographical leashes, GPS would be required and for temporal leashes, tightly synchronized clocks.
- For the purpose of encrypting the message contents, the concept of Digital Envelope [5] is used in case a public key scheme is used; otherwise the shared secret keys can be used directly.

7.2 Insider Attacker

Local Maxima We assume that the potential calculation function is so chosen that in case an adversary is absent, there are no local maxima in the system. It is possible to choose the parameters to ensure the absence of local maxima if there is no malicious node present in the network [[3], [1]]. Now, we assume the one of the mechanisms proposed in Section 6.2.1 has been implemented in order to prevent the malicious modification of the hop count. This ensures that the adversary can only increase the hop count, but not decrease it before forwarding the potential field messages. Even in case of a chain of adversarial nodes, the nodes can forward the same message without incrementing the hop count. Also due to the presence of special anycast group IDs and certificates, an adversary can not act as a sink node. Thus the adversarial node can create a sinkhole [54] and cause the packets to be routed towards it. A sinkhole is different from a local maxima as an adversary can cause messages to be routed towards itself or a particular node by false potential claims and so forth, creating a sinkhole in the process; however it would not have the highest potential among all its neighbors. A local maxima indicates that a node is not a sink, however it has a higher potential value than all its neighbors. The countermeasures for preventing a sinkhole have been suggested and will be analysed later in the section. However the adversary cannot create a local maxima in the network.

Secure Approaches Next we discuss the secure approaches which have been proposed in Section 6.3. A summary of the security mechanisms used in secure approaches has been shown in Table 6.2.

7.2.1 Approach 1

This approach uses the mechanism based on local messages to one hop neighbors, proposed in [3] in order to establish the potential field and the steepest gradient. In order to verify the potential, a node presents the messages it used in order to compute its potential, to its one hop neighbors. We discuss how this approach prevents the attacks by the attackers of various attacker classes described in Section 5.4.1.

Single adversary/multiple non colluding non neighboring adversaries

An adversary as a sender cannot indicate a higher potential than actual as it would be verified by the messages. An adversary cannot create false neighbor relations by forwarding the messages it receives from its neighbors as part of setting up the potential field as the messages are intended for one hop neighbors and the message would include the attacker's ID as the destination ID.

In order to verify the potential, a node presents the messages it used in order to compute its potential. In this approach the message which is circulated in order to establish the potential field does not contain any mutable field as it is only transmitted to the one hop neighbors, thus there is no hop count which can be modified by the adversary in order to show a higher potential. The adversary cannot modify any of the mutable fields as the signature verification would fail.

The adversary cannot reuse the message which it receives from its neighbors as part of verifying the potential as all the field in the message are immutable and the DestID field in the message includes the ID of the attacker node and it cannot be changed.

The protocol is secure against replay attacks as the TimeSlot and CTR fields are immutable and modification of any of these fields would lead to the signatures on the packet not being verified correctly. The TimeSlot field, as has been explained before, prevents the replay of a packet in a different time slot for which it has been intended. CTR prevents the replay of a packet in the same time slot as the replayed packet would be out of order from other legitimate packets.

The use of TIK, (Section 6.2.1) has been suggested in order to authenticate the packets. TIK implements packet leashes [33] which can be used in order to secure the protocol against wormholes. Packet leashes also provide hop by hop authentication, thus preventing message replay and node impersonation. However TIK requires a tight clock synchronization as a prerequisite.

An insider can not generate the potential field messages as it does not possess the certificate and the anycast group has a unique ID which, we assume, cannot be forged. Similarly, the protocol is secure against attacks where an adversarial node forges the identity of a node at a lower potential and shows its potential to be at infinity.

If an adversary does not forward the potential field establishment packets, it would not be identified as a neighbor and eventually would not be a part of the subsequent routing process. The adversary cannot flood the network with potential field messages as it cannot generate the messages and also the protocol is secure against replaying of cached messages. The adversary can also not generate false updates as the CTR in the potential field messages is an immutable field and cannot be altered. The adversary can thus not forward false routing information in the network by malicious modification of any of the immutable fields.

Here we do not discuss the problem of jamming at the physical layer. A number of countermeasures for the same have been proposed in [10] for instance, use of spread spectrum communication techniques and so forth. The problem has also been briefly discussed in Section 6.2.1

The attacker can not send the message back to the sender as the field containing the Destination ID in the message is immutable.

An adversary can not impersonate another node and generate messages with its ID as it would not possess the required keys. A countermeasure against the resource consumption attacks can be that the destination nodes after having received a number of fake packets from an insider adversary could take some action. A mobile Intrusion Detection System (IDeS) could be operating in the network.

Multiple neighboring adversarial nodes/chain

The arguments given in the case of the previous attacker class are valid for this class of attackers as well. In this case, a number of colluding nodes cannot claim to be at infinity as long as they do not have the certificate possessed by the destination. Also the destination nodes possess a special anycast group ID.

The node replication attack and sybil attack are countered by the use of any one of the schemes proposed in Section 6.2.1. If the radio fingerprinting scheme [38], discussed in Section 6.2.1 is implemented, it can counter both these attacks as well as other identity based attacks.

However the verification mechanism for potential values can be defeated by a chain of colluding nodes. In order to prevent this, an honest node which is a common neighbor to at least two of the colluding nodes can verify the potential values.

Creation of a black hole and gray hole can be thwarted by the presence of measures against misbehaving nodes as have been suggested in the previous chapter. Detection of a chain of adversarial nodes by the potential verification mechanism explained before safeguards the protocol against the attacks related to 'tunneling' of a packet between chains of adversarial nodes.

Multiple colluding nodes

All the arguments and countermeasures proposed in the previous two attacker classes are valid for this attacker class as well. Basically, a countermeasure against tunneling of messages can secure the protocol against the attacks specific to this category. Additional plausibility checks [24] for the velocity, time and distance traveled by the packet are carried out by individual nodes to determine whether the packet has been been transmitted via a tunnel or by an out of band connection. The creation of false neighbor relations via wormholes is prevented by use of packet leashes or in case TIK is not used, any other wormhole detection technique discussed in Section 6.2.1 can be implemented. As stated previously, an attacker can not claim to be at potential infinity unless it does not possess the required certification or the anycast group ID.

The potential verification mechanism can be defeated by tunneling of messages between colluding nodes. To prevent attacks related to tunneling of messages, the additional plausibility checks are employed which help to detect the tunnels and out of band connections. Thus tunneling related attacks are prevented.

7.2.2 Approach 2

This approach utilizes the broadcast mechanism for field setup and uses the approach of presenting the potential field messages as a proof for verification of potentials. Any of the mechanisms suggested in Section 6.2.1 can be used to prevent the malicious modification of the hop count.

All the arguments presented in analysis of Approach 1 are valid for this approach as well as these utilize the same mechanism for potential verification. However in this case, the potential field messages contain a mutable field, hop count. The countermeasure for preventing the malicious modification of the hop count has been described below.

Assuming that one of the security mechanisms based on hash functions as discussed in Section 6.2.1 is implemented, the adversary cannot decrease the hop count. She can not show herself to be closer to the sink than it actually is. The adversary can however increase the hop count and forward the message, but this would not create a sinkhole. The honest neighboring nodes would use the message from another node (as the potential field message is broadcast) in order to calculate the potential. Also a common neighbor of the adversarial node which increases the hop count and the previous node from which the message was received, can detect the discrepancy in the increased hop count on being presented the potential field messages for verification.

7.2.3 Approach 3

This approach uses the mechanism proposed in [3] in order to establish the field, and the cluster based approach in order to verify the potential.

In this approach we propose to use the local flooding mechanism proposed in [3] in order to establish the potential field. This approach provides the advantage of combining the first two phases as described in Section 3.2. As described in Approach 1, the message circulated in order to establish the potential field does not contain any mutable field, hence it mitigates the overhead of employing a security mechanism to prevent the malicious modification of a mutable field for instance the hop count.

The advantage that this approach has over Approach 1 is that the cluster head nodes can also serve as witness nodes which help in the detection of node replication as discussed in Section 6.2.1. An out of band connection can also be detected by a cluster head as it can monitor the communications of a node in its cluster. Similarly, the mechanisms required for

detecting a tunnel, can be used by the cluster head to determine if any of the nodes is communicating via a tunnel.

Single adversary/multiple non colluding non neighboring adversaries

A single adversarial node would be easily detected by this approach if it tries to indicate a wrong potential.

Multiple neighboring adversarial nodes/chain

A chain of nodes can be detected in the following way: If all the nodes in a chain belong to the same cluster, and the adversarial nodes try to create a sinkhole or manipulate the routing information, it will be detected by a cluster head node. If the all the nodes behave as a single entity and send a single message to the cluster, these nodes would be treated as one entity and would have the same effects as a single adversary, which can be countered. If the chain nodes are divided into different clusters, malicious modification of messages would not be possible, as the nodes which use the messages from neighbors not in the same cluster as they are, need to present the messages used for potential calculation to the cluster head.

Multiple colluding nodes

A cluster node performs the plausibility checks [24] on the routes involving the nodes in its cluster in order to detect a tunnel.

7.2.4 Approach 4

This approach combines the broadcast mechanism for the field setup with the cluster based approach in order to verify the potentials. Any of the mechanisms suggested in Section 6.2.1 can be used to prevent the malicious modification of the hop count.

This approach combines the broadcast mechanism for the field setup with the cluster based approach in order to verify the potentials. The analysis of this approach is similar to the analysis of Approach 3 as both the approaches use a similar cluster based approach in order to verify the potentials and counter other attacks. However the approach differs in the flooding mechanism used.

In this approach, the cluster head can verify the potential and also use plausibility checks to detect tunnels and out of band connections, hence any of the authentication schemes discussed in Section 6.2.1 can be used. In order to prevent the malicious hop count modification, any of the hash chain based schemes discussed in Section 6.2.1 can be used. We discuss the measures apart from the ones which have already been discussed in Approach 3.

Single adversary/multiple non colluding non neighboring adversaries

The analysis for this adversarial class is similar to the one presented in Approach 2 as both the approaches use broadcast authentication mechanism. The adversary cannot decrease the hop count, however it can forward a message without increasing the hop count. In the latter case, a cluster head would be able to detect the discrepancy in the hop count values. Any attempt at malicious modification of the mutable field would be detected by the cluster head.

Multiple neighboring adversarial nodes/chain

The chain detection mechanism is the same as that discussed in Approach 3.

Multiple colluding nodes

The cluster head node performs various plausibility checks and can also keep a tab on the routes from the nodes in their cluster in order to detect the tunneling attack.

7.2.5 Approach 5

This approach is based on tracing the route traversed by a packet in order to verify its potential. The broadcast mechanism is used in order to establish the potential field in the network and the nodes present the messages used in order to calculate their potential value for verification of potential. Additionally, TIK could be used for authentication as it implements packet leashes which safeguard the protocol against node impersonation and provide hop by hop authentication as well. Plausibility checks are used to detect a tunneled packet.

Single adversary/multiple non colluding non neighboring adversaries

In this approach, there is no mutable field in the potential field message. Hence the adversary cannot modify the potential field message. However each node is required to include its ID in the message and sign and forward it. A node cannot remove any of the previous signatures however it can include fake signatures. However, these would be verified at the time of gradient determination and the attack would be detected. A node which does not add its ID to the message and simply forwards it, does not become a part of the route used. A single adversary can thus be detected as it cannot present a message with an incorrect potential value.

The approach is secure against replay attacks due to the use of CTR and TimeSlot field as has been discussed in the previous approaches. The approach is also secure against nodes impersonating as sink nodes as the sink nodes possess special ID or certificates.

An attacker can not send the message back to the sender as the potential field message includes the node IDs as well. The attack would be detected.

As has been discussed in Approach 1 and 2, the attacks can not indicate a higher potential than it actually has due to the potential verification mechanism.

However due to each intermediate node signing the message, the overhead in this case would be greater than some of the previous approaches. Also additional mechanisms would be required in order to detect a sybil attack or node replication attack. Radio fingerprinting [38], discussed in Section 6.2.1 can be used to counter the last two attacks and other identity based attacks. Also, this approach does not detect wormholes or out of band connections, hence TIK scheme can be used for authentication , as it implements packet leashes which guard the protocol against wormholes.

Multiple neighboring adversarial nodes/chain

A chain of adversarial attackers can be detected by a common honest neighbor of at least two adjacent nodes in the chain. If the packets are 'tunneled' through the chain without any modification and forwarded by the node at the other end of the adversarial chain, it would lead to the creation of a sinkhole as the nodes would believe they are closer to the sink than they actually are. However, in order to achieve this, only some nodes in the adversarial chain would add their ID to the message and sign it. A common honest neighbor of two such adversarial nodes in a chain, one of which signs the message and the another which does not, would be able to detect the attack.

Multiple colluding nodes

Additional plausibility checks implemented by individual nodes are used to detect a tunnel.

7.3 Summary

In this chapter, we analysed the countermeasures proposed for securing the protocol against the outsider attacker. A provable security analysis for the same was also presented. Next, we analysed the five secure approaches proposed in Chapter 6, obtained by combining the various security mechanisms in order to secure the field theoretic approach. We provided an analysis of each of the approaches with respect to the three different attacker classes discussed in Section 5.4.1.

In the next chapter, we compare the different approaches and evaluate these on the basis of control packet overhead and the message size.

Chapter 8

Results

In this chapter we evaluate the performance of the approaches proposed in Section 6.3 with regards to the control overhead. A summary of the approaches has been given in Table 6.2.

8.1 Performance Evaluation

We consider the case of the single adversary/multiple non colluding adversaries. In this case, we first compare the three approaches based on broadcast mechanism, that is, Approaches 2, 4 and 5. Next, we compare the two approaches based on local flooding mechanism, that is Approaches 1 and 3. We derive the expressions for the total number of messages in the network and the message size for each of these approaches. We derive the estimates for the total number of messages exchanged per time slot, that is, the results are valid for one time slot.

Notations used In order to estimate the message overhead per node as well as the message size, we map the network as an undirected graph $G = (V, E)$, where V is the set of vertices in the graph and denotes the nodes and E indicates the set of edges in the graph, that is, the links in the network. We assume that, $n = |V|$ and $m = |E|$, that is, the total number of nodes in the network is denoted by n and the total number of links in the network is given by m. The number of neighbors of a vertex v, denoted by δ(v), is called the degree of v, where $v \in V$. The maximum degree of a vertex in a graph G is denoted by Δ. The network diameter is given by D. K denotes the number of sinks or service instances (in case of service discovery) and the number of gateways (in case of mesh networks). Throughout the discussion, we assume S to be the additional size in a message due to the signature scheme used for authentication. Also, we shall use the terms node and vertex interchangeably.

8.1.1 Broadcast mechanism

The field set up in FBR essentially takes place in two phases, establishment of potential field, where the sinks broadcast their potential value, and gradient determination, where nodes exchange messages with their neighbors regarding their potential values in order to determine the steepest gradient as discussed in Section 3.2. For the purpose of calculating the number of messages exchanged, we consider the two phases separately for broadcast mechanism.

Potential field Establishment In the first phase, each sink broadcasts a message to each of its neighbors which forward the message. Each node sends a message to each of its neighbors except the one from which the message was received. Thus, every node v sends $\delta(v) - 1$ messages. Therefore, considering the maximum number of messages which can be sent by every node, the total number of messages sent by all the nodes would be of the order of $n\Delta$, as there are n nodes in the network, that is

Broadcast mechanism: Potential field establishment

$$\text{Total number of messages} \in O(n\Delta) \tag{8.1}$$

In the second phase, each node sends a message to each of its neighbors, indicating its potential value. Therefore, a node v sends $\delta(v)$ messages. Considering the messages sent by every node in the network, the total number of messages in this phase would be of the order of $n\Delta$, that is,

Broadcast mechanism: Gradient determination

$$\text{Total number of messages } \in O(n\Delta) \tag{8.2}$$

Message size In the first phase, the size of each message is constant, that is,

Broadcast mechanism: Potential field establishment

$$\text{Message size } \in O(1) \tag{8.3}$$

In the second phase, we consider the size of the messages exchanged to be constant, that is,

Broadcast mechanism: Gradient determination

$$\text{Message size } \in O(1) \tag{8.4}$$

Considering the two phases described above, assuming $k = 1$, the total number of messages and message size for each of the two phases is given in Table 8.1

Phase	Total number of messages	Message size
Establishment of potential field	$O(n\Delta)$	$O(1)$
Gradient determination	$O(n\Delta)$	$O(1)$
Total	$O(2n\Delta)$	$O(1)$

Table 8.1: Total number of Messages and Message size for broadcast mechanism

The field theoretic approach has been analysed for use in anycast routing and service discovery in [6]. For considering the variation in the message overhead with the number of sinks or destination nodes, we base our evaluation on the basic results which have been derived from the performance of the protocol for service discovery by V. Lenders in [6]. It has been shown that using the broadcast mechanism proposed in [1], the control overhead traffic rate, which includes the messages flooded throughout the network by the sinks (services) as well as messages exchanged for determination of steepest gradient, depends linearly on the number of sinks (service instances), that is,

Broadcast mechanism with k sinks (without flooding reduction techniques): Total messages

$$\text{Total number of messages } \in O(2kn\Delta) \tag{8.5}$$

where k is the number of sinks or destination nodes in the network.

However, it has also been observed that if we use the flooding reduction techniques proposed in [6], the control overhead per node measured as the average number of control packets sent does not increase as the number of sinks (services) increases. This is because the flooding reduction techniques propose the caching of advertisements or broadcast messages from the service instances or sinks of the same type before forwarding them. Another flooding reduction technique used: the advertisements or messages which do not significantly alter the potential field are not propagated. Therefore, when the flooding reduction techniques are used, the total number of messages is independent of the number of sinks, that is,

Broadcast mechanism with k sinks (using flooding reduction techniques): Total messages

$$\text{Total number of messages} \in O(2n\Delta) \tag{8.6}$$

where k is the number of sinks or destination nodes in the network

Based on these observations, we compare the performance of the three approaches which use the broadcast mechanism proposed in [6].

Local verification by showing messages: Approach 2

Considering the case of a single adversary, the additional overhead incurred at every node by applying this approach to the service discovery scheme proposed in [1], would include the overhead due to the authentication mechanism (signing and verifying the signature etc.) and the mechanism used for preventing the hop count modification. However with reference to the number of messages which are exchanged between the nodes in the original approach, the message size would increase as the nodes need to transmit the messages used for potential calculation to their neighbors as well. However the number of control packets per node would remain unchanged. We have also proposed the addition of two new fields *TimeSlot* and *CTR*, which eliminate the need for *Seq No.* and *TTL* field as suggested in the original protocol. Based on the results from [6] as discussed above, the control overhead per node would increase linearly with the number of sinks in the absence of the flooding reduction techniques and would remain constant as the number of sinks increases if the overhead reduction techniques are employed. The message size would increase as a result of the signature scheme used for authentication as well. Therefore, assuming the use of flooding reduction techniques,

Broadcast mechanism with local verification: Potential field establishment

$$\text{Total number of messages} \in O(n\Delta) \tag{8.7}$$

Broadcast mechanism with local verification: Gradient determination

$$\text{Total number of messages} \in O(n\Delta) \tag{8.8}$$

Broadcast mechanism with local verification: Total messages

$$\text{Total number of messages} \in O(2n\Delta) \tag{8.9}$$

Message size The message size in this approach would however be more than the message size in the basic broadcast mechanism with no potential verification. In the first phase when the sinks broadcast the messages containing their potential value, the messages are signed by the node. Thus the message size would increase by S, that is,

Broadcast mechanism with local verification: Potential field establishment

$$\text{Message size} \in O(S) \tag{8.10}$$

During the second phase, the message size is further increased as each node also sends the messages used by it for computation of the potential value. Each node v sends the messages received from its neighbors $\delta(v) - 1$ along with its own potential value. Therefore, the message size would be of the order of:

Broadcast mechanism with local verification: Gradient determination

$$\text{Message size} \in O(S\Delta) \tag{8.11}$$

A summary of the results for the local verification mechanism has been given in Table 8.2

Phase	Total number of messages	Message size
Establishment of potential field	$O(n\Delta)$	$O(S)$
Gradient determination	$O(n\Delta)$	$O(S\Delta)$
Total	$O(2n\Delta)$	

Table 8.2: Total number of Messages and Message size for broadcast mechanism with local verification: Approach 2

Cluster approach: Approach 4

Considering the case of a single adversary, additional overhead would be incurred due to the cluster formation and the cluster head election algorithms. The cluster formation algorithm used in [44] forms a cluster consisting of only one hop neighbors. Other cluster based algorithms have also been discussed. Assuming that the clusters which have been formed include nodes which are only one hop away from the cluster node, we estimate the control overhead per node. After the cluster head election, each node sends a message to the cluster head indicating its potential. If a node uses messages from its neighbors which are not in the same cluster, it also sends the messages received from these neighbors to the cluster head. In this process, only the message size would increase, however the total number of messages (excluding the cluster formation and cluster head election) would be the same as that used for verification of potential by showing messages as in previous approach.

For instance, consider a cluster of c nodes where one node is the cluster head. Each node sends a message to its neighbor in order to determine the steepest gradient. Each node also sends a message to the cluster head. During the slot in which a node is elected as the cluster head, it receives messages from all $c - 1$ nodes and makes the potential verifications. Thus, effectively, there is an additional message sent by every node to the cluster head as compared to previous approach where nodes perform local verifications. However, there would be one less node whose potential would need to be verified as the cluster head would perform all the verifications itself. So the number of messages which are exchanged for the purpose of verification, is the same as in the case of local verifications (Section 8.1.1). However there are additional messages circulated in the network for formation of clusters and cluster head election. By adjusting the duration after which the reformation and re-election takes place, the overhead can be reduced. Increasing the duration would, however, also make the protocol prone to attacks in the event that an adversarial node is elected as a cluster head.

For the performance evaluation of the cluster approach, we consider several phases in which the approach proceeds.

Broadcast The number of message sent by by every node in the initial phase is the same as in the case of local verification as each sink broadcasts the message and every node forwards the message to each of its neighbors except the neighbor from whom the message was received. Therefore,

Broadcast mechanism with cluster approach: Potential field establishment

$$\text{Total number of messages } \in O(n\Delta) \tag{8.12}$$

Cluster formation We consider the formation of a one hop cluster. Several cluster formation algorithms have been proposed in literature; we propose a general analysis for a basic cluster formation algorithm from the point of view of messages exchanged. We give a lower bound for the number of messages. Each node sends minimum one message to every neighbor for the establishment of a cluster. Initially all nodes are in single node clusters. Therefore, each node sends $\delta(v)$ messages. Computing for n nodes in the network, the total number of messages sent are of the order of $n\Delta$. Therefore,

Broadcast mechanism with cluster approach: Cluster formation

$$\text{Total number of messages } \in O(n\Delta) \tag{8.13}$$

Cluster head election After clusters are formed, a cluster head is elected via local computations within the clusters. Each node sends minimum one message to its neighbors, therefore each node sends $\delta(v)$ messages. Messages are also involved in the local computations for the leader election. Hence,

Broadcast mechanism with cluster approach: Cluster head election

$$\text{Total number of messages} \in O(2n\Delta) \tag{8.14}$$

Gradient determination For the purpose of potential verification during the gradient determination phase, each node in a cluster sends a message to every node within the cluster except the cluster head indicating its potential value. Additionally, each node also sends a message to the cluster head for verification. A node also appends the messages used for potential calculation if it used messages from any of its neighbors which lie outside the cluster.

The number of messages sent by a node during this phase would be the same as in the local verification approach. Therefore,

Broadcast mechanism with cluster approach: Gradient determination

$$\text{Total number of messages} \in O(n\Delta) \tag{8.15}$$

Message size The message size in the initial phase when messages are broadcast in the network is the same as in the case of local verification. Therefore,

Broadcast mechanism with cluster approach: Potential field establishment

$$\text{Message size} \in O(S) \tag{8.16}$$

We consider the size of the message circulated during the cluster formation and cluster election to be constant. However the messages are signed by the nodes for the purpose of authentication. Therefore,

Broadcast mechanism with cluster approach: Cluster formation

$$\text{Message size } \in O(S) \tag{8.17}$$

Broadcast mechanism with cluster approach: Cluster head election

$$\text{Message size} \in O(S) \tag{8.18}$$

During the gradient determination phase, the messages sent by the nodes to their neighbors, except the cluster head are of a constant size, that is,

$$\text{Message size} \in O(S) \tag{8.19}$$

The messages sent by the node to the cluster head, also include the messages used by the node for potential computation. In the extreme case, a node v would need to send the messages received from all the $\delta(v) - 1$ neighbors to the cluster head. Therefore, in that case,

Broadcast mechanism with cluster approach: Gradient determination

$$\text{Message size} \in O(S\Delta) \tag{8.20}$$

A summary of the results for the cluster approach has been given in Table 8.3.

Phase	Total number of messages	Message size
Establishment of potential field	$O(n\Delta)$	$O(S)$
Cluster formation	$O(n\Delta)$	$O(S)$
Cluster head election	$O(2n\Delta)$	$O(S)$
Potential verification	$O(n\Delta)$	$O(S\Delta)$
Total	$O(5n\Delta)$	

Table 8.3: Total number of Messages and Message size for broadcast mechanism with cluster approach: Approach 4

Tracing packet route: showing messages: Approach 5

The additional overhead in this case would be due to the aggregate signature scheme employed. Comparing it with the local verification approach, there would be no requirement to employ a mechanism for hop count modification as the hop count is eliminated in this approach. However the employment of a secure aggregate signature scheme would increase the message size and overhead per node. The number of control packets per node would be the same as in the original approach for broadcast mechanism. Therefore,

Broadcast mechanism with tracing packet route: Potential field establishment

$$\text{Total number of messages } \in O(n\Delta) \tag{8.21}$$

Broadcast mechanism with tracing packet route: Gradient determination

$$\text{Total number of messages} \in O(n\Delta) \tag{8.22}$$

Broadcast mechanism with tracing packet route: Total messages

$$\text{Total number of messages } \in O(2n\Delta) \tag{8.23}$$

As this approach uses aggregate signature schemes, the potential field message would include a list of the nodes on the route traversed by the packet. Therefore, the message size would increase in the first phase as node IDs would be appended to the message as well. There would be no hop count field in the potential field establishment message. Therefore, considering that the additional overhead due to the addition of a node ID to the message is of the order of $\log n$,

Broadcast mechanism with tracing packet route: Potential field establishment

$$\text{Message size} \in O(SD \log n) \tag{8.24}$$

During the gradient determination phase, each node would show the signed messages received from its neighbor, the same procedure is followed as in the case of local verification.

Broadcast mechanism with local verification: Gradient determination

$$\text{Message size} \in O(S\Delta) \tag{8.25}$$

A summary of the results for the packet tracing approach has been given in Table 8.4.

Phase	Total number of messages	Message size
Establishment of potential field	$O(n\Delta)$	$O(SD \log n)$
Gradient determination	$O(n\Delta)$	$O(S\Delta)$
Total	$O(2n\Delta)$	

Table 8.4: Total number of Messages and Message size for broadcast mechanism with packet tracing approach: Approach 5

8.1.2 Comparison: Approach 2, 4, 5

For the case of single adversary, the cluster approach would incur the highest control overhead due to the additional cluster formation and cluster election messages which would be circulated in the network. However this can be reduced to a great extent by using efficient algorithms for the same. The local verification approach, where each node verifies its neighbor's potential through the potential field messages shown as proof, would incur the least overhead among the three schemes, provided the mechanism used for preventing malicious hop count modification is efficient. The approach based on tracing the packet route incurs the same additional overhead as the local verification approach with respect to the control packets per node, however the overhead per node would depend on the aggregate signature scheme used and the verifications which are required to be performed. In the local verification and packet tracing approach, the message size would increase as well. The number of messages and message sizes, as estimated above for all three approaches as well as the original broadcast mechanism without using any security mechanism have been summed up in Table 8.5.

If we consider the case of a stronger adversary, that is, multiple attackers in a chain or colluding nodes, additional measures would need to be employed in all the three approaches. The difference would be that in the cluster approach, proposed in Approach 4, only the cluster head can perform the verifications, for instance, the plausibility checks and interacting with the cluster head of a neighboring cluster. The cluster heads can also act as witness nodes [39] as discussed in Section 6.2.1 in order to mitigate the node replication attack. The cluster nodes can also act in cooperation with an Intrusion detection system [44] and detect and prevent other attacks on the protocol as well. There are other schemes which can be employed, for instance, radio fingerprinting [38] and secure localization techniques can be adapted to the protocol such that, the cluster head performs the bulk of verifications. It would indicate additional overhead at the cluster head for one time slot, but it would decrease the control traffic overhead for other nodes in the cluster.

For the local verification and packet tracing approach, proposed in Approach 2 and 5 respectively, additional techniques would need to be applied as discussed in Section 6.3. These verification would be performed at every node and would also involve exchange of messages between the nodes for further verifications, thus increasing the control traffic overhead per node.

Approach	Phase	Total number of messages	Message size
Broadcast mechanism			
	Establishment of potential field	$O(n\Delta)$	$O(1)$
	Gradient determination	$O(n\Delta)$	$O(1)$
	Total	$O(2n\Delta)$	
Broadcast mechanism with local verification: Approach 2			
	Establishment of potential field	$O(n\Delta)$	$O(S)$
	Gradient determination	$O(n\Delta)$	$O(S\Delta)$
	Total	$O(2n\Delta)$	
Broadcast mechanism with cluster approach: Approach 4			
	Establishment of potential field	$O(n\Delta)$	$O(S)$
	Cluster formation	$O(n\Delta)$	$O(S)$
	Cluster head election	$O(2n\Delta)$	$O(S)$
	Potential verification	$O(n\Delta)$	$O(S\Delta)$
	Total	$O(5n\Delta)$	
Broadcast mechanism with packet tracing approach: Approach 5			
	Establishment of potential field	$O(n\Delta)$	$O(SD\log n)$
	Gradient determination	$O(n\Delta)$	$O(S\Delta)$
	Total	$O(2n\Delta)$	

Table 8.5: Total number of Messages and Message size for broadcast mechanism: Summary

The cluster approach can be used in the network in case of a stronger adversary, with less overhead as compared to the other two approaches, depending on the efficiency of the algorithms applied for cluster formation and cluster head election and the combining of novel security mechanisms with the cluster approach.

8.1.3 Local field set up mechanism

The field theoretic approach has also been analysed for use in wireless mesh networks in [7]. The approach used for establishing the potential field is based on local exchange of potential messages and is thus scalable for large networks. As compared to other routing protocols for mesh networks, the routing overhead, that is, the routing packets for the field establishment per second per node, remain relatively unaffected as the number of gateways (sinks) increase. This is because the approach uses local beacons instead of a global flooding mechanism. The arguments proposed in Section 8.1.1 and the comparison between the local verification of potential values and cluster approach are valid for the local field set up mechanism proposed in [3] as well in terms of the additional overhead imposed due to the secure approaches.

In the original protocol proposed in [7], each node sends a message to its neighboring nodes at least once. After calculating the potential value from the received messages. Therefore, each node v sends a message to its $\delta(v) - 1$ neighbors. Therefore the total number of messages sent by every node, would be of the order of Δ. Therefore,for n nodes,

Local field set up mechanism: Potential field establishment and Gradient determination

$$\text{Total number of messages} \in O(n\Delta) \tag{8.26}$$

The analysis for the local verification approach (Table 8.2) and the cluster approach (Table 8.3) is the same as in the previous section. The results have been summed up in Table 8.6.

Approach	Phase	Total number of messages	Message size
Local field set up mechanism	Field set up	$O(n\Delta)$	$O(1)$
Local field set up mechanism with local verification: Approach 1	Field set up	$O(n\Delta)$	$O(S\Delta)$
Local field set up mechanism with cluster approach: Approach 3	Cluster formation Cluster head election Field set up and potential verification Total	$O(n\Delta)$ $O(2n\Delta)$ $O(n\Delta)$ $O(4n\Delta)$	$O(S)$ $O(S)$ $O(S\Delta)$

Table 8.6: Total number of Messages and Message size for local field set up mechanism: Summary

8.2 Summary

In this chapter, we evaluated the secure approaches in terms of control packet overhead and the message size in every phase. We compared the different approaches and the performance evaluation was presented for both the flooding mechanisms: Global broadcast and local flooding mechanism.

Chapter 9

Conclusions and Future Work

In this chapter, we conclude our work on securing the field theoretic approach. We first summarize our main results and list the aspects studied in the preceding chapters. The second section contains our recommendations for the future work in the field of secure field based routing.

9.1 Summary

In this project, we identified various attacks which can be mounted on the field theoretic approach. We reviewed the attacks on other ad hoc routing protocols in general and studied their applicability to the field theoretic approach. We studied the basic features of field theoretic approach and identified three distinct phases in the working of the protocol, that is, establishment of potential field, gradient determination and packet forwarding.

Based on the various attacks, we distinguished the attacker as being an outsider or an insider attacker, in accordance with the information available to the attacker and the type of attacks it can mount on the system. We also described the attacker model and listed the major security objectives we are working towards. We described the different attacks which can be mounted by these attackers on the field theoretic approach and described the impact of these attacks on the protocol. With reference to the field theoretic approach, we further proposed a novel taxonomy for classifying the attacker classes for the insider attacker. Namely, *Single adversary/multiple non colluding non neighboring adversaries, Multiple neighboring adversarial nodes/chain of adversarial nodes and Multiple colluding nodes.*

We reviewed various security mechanisms that have been proposed in literature in order to secure the protocols against specific attacks and also proposed two novel techniques for potential verification: local verification of potential by showing the messages used in order to compute the potential as a proof and the cluster approach.

Based on the available security mechanisms and the potential verification techniques, we proposed five different approaches which can be used to secure the field theoretic approach. We analysed these approaches and gave the performance evaluation in terms of control packet overhead and message size.

9.2 Future Work

There are several directions for further work. Firstly, the security objectives we strived to achieve by means of the secure approaches, are general in nature and apply to most of the scenarios. However each application has certain specific objectives which need to be achieved. For instance, for sensor networks, location based attacks would also need to be countered and the process of data aggregation would need to be secured as well. Any of the

proposed approaches can thus be adapted for the application and the additional objectives can be met using the general solutions we reviewed in Section 6.2.

Secondly, the approaches we proposed mainly secure the field set up phase of FBR. The attacks on packet forwarding are similar in nature to those which were discussed in Section 5.2. The solutions can be chosen in accordance with the requirement of the situation, for instance, availability of a central point like the base station in sensor networks or efficiency issues. The use of threshold cryptography can also be considered.

The metric suggested for distance estimation in field based routing originally was hop count. The authors in [6] have suggested considering the use of other metrics for the same. This might mitigate the need for certain measures as well and make the protocol less prone to attacks based on mutable information. Also, there have been advances in the field of several security mechanisms which can secure the protocol against various attacks simultaneously, for instance, radio fingerprinting can safeguard the protocol against node impersonation, sybil attack or node replication. However the implementation of these measures is still in the development stage and the infrastructure requirements need to be considered as well. Such mechanisms can be integrated with the secure approaches proposed in Section 6.3 to provide a secure protocol as well.

Bibliography

[1] Vincent Lenders, Martin May, Bernhard Plattner. Service Discovery in Mobile Ad Hoc Networks. A Field Theoretic Approach. *Elsevier Journal on Pervasive and Mobile Computing,* Elsevier, Vol. 1, No. 3, pages 343-370, September, 2005.

[2] Praveen Kumar, Joy Kuri, Pavan Nuggehalli, Mario Strasser, Martin May, Bernhard Plattner. Connectivity-aware Routing in Sensor Networks. *In Proceedings of Sensorcomm 2007,* IEEE, September, 2007.

[3] Rainer Baumann, Simon Heimlicher, Vincent Lenders and Martin May. HEAT: Scalable Routing in Wireless Mesh Networks Using Temperature Fields. *In Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM),* Helsinki, Finland, June 2007.

[4] Vincent Lenders, Martin May, Bernhard Plattner. Density-based Anycast. A Robust Routing Strategy for Wireless Ad Hoc Networks. In *Proceedings of IEEE/ACM Transactions on Networking,* IEEE, 2007.

[5] Levente Buttyán and Jean-Pierre Hubaux. *Security and Cooperation in Wireless Networks, Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing,* Cambridge University Press.

[6] Vincent Lenders, PhD thesis. *Field-based Routing and its Application to Wireless Ad Hoc Networks* Shaker Verlag, 2006.

[7] Rainer Baumann, Dissertation at ETH Zurich, *Building Scalable and Robust Wireless Mesh Networks,* ETH No. 17306, TIK-Schriftenreihe Nr. 93, Shaker Verlag, 2007.

[8] Elizabeth M. Belding Royer. Routing Approaches in Mobile ad hoc Networks. In *Mobile Ad Hoc Networking,* edited by Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic, IEEE , 2004.

[9] Larry L. Peterson and Bruce S. Davie. *Computer Networks-A Systems Approach, 3rd edition,* Morgan Kaufmann Publishers, 2003.

[10] R. A. Poisel. *Modern Communications Jamming Principles and Techniques.* Artech House Publishers, 2006.

[11] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications,* WMCSA '99.

[12] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, a. Qayyum et L. Viennot. Optimized Link State Routing Protocol. In *IEEE INMIC* Pakistan, 2001.

[13] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing (ed. T. Imielinski and H. Korth),* Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[14] Yih-Chun Hu , Adrian Perrig, and Dave Johnson. Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks. In *Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking (ACM Mobicom),* Atlanta, Georgia, September 23 - 28, 2002.

[15] M. G. Zapata and N. Asokan. Securing ad hoc routing protocols. In *Proceedings of the 1st ACM Workshop on Wireless Security* (Atlanta, GA, USA, September 28 - 28, 2002). WiSE '02. ACM, New York, NY, 1-10.

[16] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, Elizabeth M. Belding-Royer. A Secure Routing Protocol for Ad Hoc Networks. *icnp*, p. 78, *10th IEEE International Conference on Network Protocols (ICNP'02),* 2002.

[17] Helena Rifa-Pous and Jordi Herrera-Joancomarti. Secure Dynamic MANET On-demand (SEDYMO) Routing Protocol. In *Proceedings of the Fifth Annual Conference on Communication Networks and Services Research (May 14 - 17, 2007)*. Communications Networks and Services Research Conference. IEEE Computer Society, Washington, DC, 372-380.

[18] Azzedine Boukerche and Yonglin Ren. ARMA: An Efficient Secure Ad Hoc Routing Protocol. In *Proceedings of Global Telecommunications Conference, GLOBECOM '07,* IEEE, 2007.

[19] L. Butty´an and I. Vajda. Towards provable security for ad hoc routing protocols. In *Proceedings of the ACM Workshop on Security in Ad Hoc and Sensor Networks (SASN),* October 2004.

[20] Gergely Ács, Levente Buttyán, István Vajda. Provably Secure On-Demand Source Routing in Mobile Ad Hoc Networks.In *IEEE Transactions on Mobile Computing,* vol. 5, no. 11, Nov 2006.

[21] Yih-Chun Hu, David B. Johnson, and Adrian Perrig. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless AdHoc Networks. In *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2002),* IEEE Press, 2002.

[22] P. Papadimitratos and Z.J. Haas. Secure Routing for Mobile Ad Hoc Networks. In *Proceedings of SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002),* Jan. 2002.

[23] Lin Chen, Jean Leneutre, Jean-Jacques Puig. A Secure and Efficient Link State Routing Protocol for Ad Hoc Networks. In *International Conference on Wireless and Mobile Communications, 2006. ICWMC '06.*

[24] C. Harsch, A. Festag, and P. Papadimitratos. Secure Position-Based Routing for VANETs. In *2007 IEEE 66th Vehicular Technology Conference (VTC 2007),* Baltimore, USA, 30 September - 3 October 2007.

[25] D. Raffo, C. Adjih, T. Clausen, and P. Mühlethaler. An Advanced Signature System for OLSR. In *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks* (Washington DC, USA, October 25 - 25, 2004). SASN '04. ACM, New York,

[26] B. Schneier. Attack Trees. In *Dr Dobbs Journal. December 1999.* http://www.schneier.com/paper-attacktreesddj-ft.html

[27] J.R. Douceur. The Sybil attack. *In Proceedings of 1st International Workshop on Peer-to-Peer Systems (IPTPS 02),* 2002.

[28] J. Newsome, E. Shi, D. Song and A. Perrig. The sybil attack in sensor networks: analysis & defenses. In *Proceedings of the Third international Symposium on information Processing in Sensor Networks* (Berkeley, California, USA, April 26 - 27, 2004). IPSN '04. ACM, New York,

[29] Y. Hu,A. Perrig, and D.B. Johnson. Rushing attacks and defense in wireless ad hoc network routing protocols. In *Proceedings of the 2nd ACM Workshop on Wireless Security, WiSe '03,* San Diego, CA, USA, September 19 - 19, 2003. ACM, New York, NY, 30-40.

[30] Lei Guang and Chadi Assi. Vulnerabilities of ad hoc network routing protocols to MAC misbehavior. In *Proceedings of IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob'2005),* 2005.

[31] Jeanette Tsang and Konstantin Beznosov. A Security Analysis of the Precise Time Protocol. *Technical report LERSSE-TR-2006-02*. http://lersse-dl.ece.ubc.ca/

[32] Mario Strasser, Christina Pöpper, Srdjan Capkun, Mario Cagalj. Jamming-resistant Key Establishment using Uncoordinated Frequency Hopping. In *Proceedings of the IEEE Symposium on Security and Privacy,* Oakland, California, USA, May, 2008.

[33] Hu, Yih-Chun, Adrian Perrig, and Dave Johnson. Packet Leashes:A Defense against Wormhole Attacks in Wireless Networks. In *Proceedings of IEEE Infocom,* San Francisco, California, March 30 - April 3, 2003.

[34] Yih-Chun Hu, Adrian Perrig, David B. Johnson. Efficient Security Mechanisms for Routing Protocols. *Proceedings of the Network and Distributed System Security Symposium, NDSS 2003, San Diego, California, USA*, NDSS, The Internet Society, 2003,

[35] T. Aura, "Cryptographically generated addresses (CGA)". In *RFC 3972,* 2005.

[36] V.N. Padmanabhan and D. R. Simon. Secure Traceroute to Detect Faulty or Malicious Routing. In *ACM SIGCOMM Workshop on Hot Topic in Networks (HotNets-I),* October 2002.

[37] Adrian Perrig, Ran Canetti, J.D. Tygar, and Dawn Xiaodong Song. Efficient authentication and signing of multicast streams over lossy channels. In *IEEE Symposium on Security and Privacy,* May 2000.

[38] J. Hall, M. Barbeau and E. Kranakis. Enhancing Intrusion Detection in Wireless Networks Using Radio Frequency Fingerprinting. *Communications, Internet and Information Technology (CIIT),* St. Thomas, US Virgin Islands, November 22-24, 2004.

[39] Bryan Parno, Adrian Perrig and Virgil Gligor. Distributed Detection of Node Replication Attacks in Sensor Networks. *Proceedings of the 2005 IEEE Symposium on Security and Privacy,* May 8-11, 2005, Oakland, CA.

[40] Adrian Perrig and Robert Szewczyk and J. D. Tygar and Victor Wen and David E. Culler. SPINS: security protocols for sensor networks. *Wireless Networks,* Volume 8, no. 5, Sep 2002.

[41] R. L. Rivest and A. Shamir and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. In *Commun. ACM*, volume 26, no 1, 1983, ACM, New York.

[42] J. Arkko, J. Kempf, B. Zill, P. Nikander. "SEcure Neighbor Discovery (SEND)". In *RFC 3971,* 2005.

[43] C. Schnorr. Efficient Signature Generation by Smart Cards. In *Journal of Cryptology,* Vol.4, 1991.

[44] Y. Huang and W. Lee. A cooperative intrusion detection system for ad hoc networks. In *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks (Fairfax, Virginia). SASN '03.* ACM, New York.

[45] Y. Zhang and W. Lee. Intrusion detection in wireless ad hoc networks. *ACM MOBICOM,* 2000.

[46] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th Annual international Conference on Mobile Computing and Networking* (Boston, Massachusetts, United States, August 06 - 11, 2000). MobiCom '00. ACM, New York.

[47] A. Wood and J. Stankovic. Denial of Service in Sensor Networks. *IEEE Computer ,* Vol. 35, No. 10, October 2002.

[48] T. Leinmuller, C. Maihofer, E. Schoch, and F. Kargl. Improved Security in Geographic Ad Hoc Routing Through Autonomous Position Verification. In *Proceedings of the 3rd international Workshop on Vehicular Ad Hoc Networks* (Los Angeles, CA, USA, September 29 - 29, 2006). VANET '06. ACM, New York.

[49] Wensheng Zhang, N. Subramanian and Guiling Wang. Lightweight and Compromise-Resilient Message Authentication in Sensor Networks. In *IEEE INFOCOM 2008. The 27th Conference on Computer Communications.* April 2008.

[50] Saurabh Ganeriwal, Christina Pöpper, Srdjan Capkun, Mani B. Srivastava. Secure Time Synchronization in Sensor Networks. Accepted for publication in *ACM Transactions on Information and System Security,* 2008.

[51] R.C. Merkle. A Digital Signature Based on a Conventional Encryption Function. In *A Conference on the theory and Applications of Cryptographic Techniques on Advances in Cryptology* (August 16 - 20, 1987). C. Pomerance, Ed. Lecture Notes In Computer Science, vol. 293. Springer-Verlag, London.

[52] Cynthia Kuo, Mark Luk, Rohit Negi, and Adrian Perrig. Message-In-a-Bottle: User-Friendly and Secure Key Deployment for Sensor Nodes. In *Proceedings of the ACM Conference on Embedded Networked Sensor System (SenSys 2007),* Sydney, Australia. November 6 - 9, 2007.

[53] Yih-Chun Hu and Adrian Perrig. A Survey of Secure Wireless Ad Hoc Routing. In *IEEE Security & Privacy, special issue on Making Wireless Work,* 2(3). May/June 2004.

[54] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. In *Ad hoc Networks Journal (Elsevier),* 1(2-3). Sept. 2003.

[55] L. Zhou and Z. J. Haas. Securing Ad Hoc Networks. *IEEE Network Magazine,* vol. 13, no.6, November/December 1999.

[56] I.F. Akyildiz and Xudong Wang. Survey on Wireless Mesh Networks. In *Communications Magazine, IEEE,* Sept. 2005, Volume: 43, Issue: 9.

[57] E. Shi and A. Perrig. Designing secure sensor networks. In *Wireless Communications, IEEE,* Dec. 2004, Volume: 11, Issue: 6.

[58] Frank Kargl, Stefan Schlott, Andreas Klenk, Alfred Geiss, Michael Weber. Securing Ad hoc Routing Protocols. In *Euromicro, 30th EUROMICRO Conference (EUROMICRO'04)*, 2004.

[59] P. Argyroudis and D. O'Mahony. Secure routing for mobile ad-hoc networks. In *IEEE Communications Surveys and Tutorials,* Series 7, (3), 2005.