AD HOC WIRELESS NETWORKING
X. Cheng, X. Huang and D.-Z. Du (Eds.) pp. 319 - 364

A Survey of Wireless Security in Mobile Ad Hoc Networks: Challenges and Available Solutions [1]

Wenjing Lou and Yuguang Fang
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611
E-mail: {wjlou@,fang@ece.}ufl.edu

Contents

1 Introduction **320**

2 Security Challenges **322**
2.1 Security Attacks 322
2.2 Security Mechanisms 324
2.3 Security Services 325

3 Available Security Solutions **327**
3.1 Defending Against Physical Attacks 327
3.2 Enforcing Confidentiality 328
3.2.1 Hiding the Nodes' Transmissions 328
3.2.2 Securing the Communications Path 328
3.2.3 Enhancing Confidentiality via Multipath Routing 329
3.2.4 Preventing Traffic Analysis 332
3.3 Key Management 333
3.4 Authentication 334
3.5 Protecting Routing Protocols 335
3.6 Handling Node Misbehavior 336
3.6.1 Network Layer Misbehavior 336
3.6.2 MAC Layer Misbehavior 338
3.7 Intrusion Detection 339

[1] This work was supported in part by the Office of Naval Research under Young Investigator Award N000140210464 and under grant N000140210554, and the NSF under Faculty Early Career Development Award ANI-0093241 and under grant ANI-0220287.

4 Key Management **341**
4.1 Fundamentals and Overview 342
4.2 Distributed and Cooperative Certificate Authority 345
4.3 Self-organized Public Key Management 346

5 Secure Routing Protocols **348**
5.1 Vulnerabilities Analysis of Mobile Ad Hoc Routing Protocols 349
5.1.1 Modification of Routing Information 349
5.1.2 Fabrication of Routing Information 350
5.1.3 Replay 351
5.1.4 Denial of Service (DOS) 353
5.2 Protection of Routing Protocols 353
5.2.1 Hop-by-hop Authentication 353
5.2.2 End-to-end Authentication 354
5.2.3 Reactive Approach 355

6 Conclusions **357**

References

1 Introduction

A mobile ad hoc network (MANET) is a self-configurable, self-organizing, infrastructureless multi-hop wireless network. By self-configurable and self-organizing, we mean that an ad hoc network can be formed, merged together or partitioned into separated networks on the fly depending on the networking needs, and few administrative actions need to be performed for network setup and maintenance. By infrastructureless, we mean that an ad hoc network can be promptly deployed without relying on any existing infrastructure such as base stations for wireless cellular networks. By multi-hop wireless, we mean that in an ad hoc network the routes between end users may consist of multi-hop wireless links, as compared to the single wireless hop in a wireless LAN or a cellular network, where only the last hop, e.g. from the end user to the access point or the base station, is wireless, all the links beyond that point remain wired. In addition, each node in a mobile ad hoc network is capable of moving independently and forwarding packets to other nodes. The rapidly deployable and self-organizing features make mobile ad hoc networking very attractive in military applications, where fixed infrastructures are not available or reliable, and fast network establishment and self-reconfiguration are necessary. Primary applications of mobile ad

hoc networks include the tactical communications in battlefields and disaster rescue after an earthquake, for example, where the environments are hostile and the operations are security-sensitive, yet fast and reliable deployments are a must. Recently, due to the availability of wireless communication devices that operate in the ISM (Industrial, Scientific and Medical) bands and other unlicensed band, the interest in mobile ad hoc networks has been extended to civilian life such as on-the-fly setup for conferencing and home-area wireless networking.

Although mobile ad hoc networks have attracted tremendous attention in the last few years, most research efforts have been focused on the development of the network architecture itself, particularly in the network routing protocol and medium access control (MAC) protocol design. We observe that relatively little works have been carried with the security consideration. Lessons we learned from the recent history of the Internet and cellular networks tell us that if a given network architecture is not designed with security from the very start, the security vulnerabilities will be exploited by malicious users, and the network might be paralyzed by various types of attacks. Moreover, addressing security issues as an after thought can be very painful, expensive, and also inefficient ([24]). Thus, incorporation of security aspects into the currently formalized ad hoc networking architecture is of paramount importance.

Computer network and information security has been extensively studied in the wired Internet context in the past. A number of effective security mechanisms are already in place. However, due to the salient (e.g. infrastructureless, wireless, mobile, self-organizing) features of mobile ad hoc networks, the security approaches that are valid in the Internet may not be fully applicable in mobile ad hoc networks. Many new challenges that restrict the applicability of security mechanisms for this new environment arise. First of all, the wireless channels suffer from poor protection and are more susceptible to various forms of attacks such as passive eavesdropping, active signal interference, and jamming. Secondly, most ad hoc network routing protocols are co-operative in nature and rely on an implicit trust relationship among participating nodes to route packets. The co-operative nature makes it much easier for data tampering, impersonation, and denial of service (DoS) attacks. Thirdly, the lack of a fixed infrastructure and a central concentration point makes some conventional security mechanisms difficult to apply. For example, it makes it difficult for an intrusion detection system to collect audit data, and also impedes the deployment of wide spread asymmetric cryptography due to the lack of a PKI (Public Key Infrastructure), where a centralized certificate authority is needed. Fourthly,

mobile devices tend to have limited memory, slow processing, low battery power, as well as finite radio transmission bandwidth, which limit the practical deployment of computationally intensive or more comprehensive security schemes in MANET environments. Finally, continuous and unpredictable ad hoc network mobility clouds the distinction between normalcy and anomaly, thus makes the detection of malicious behaviors difficult.

In this chapter, we focus on various security issues in mobile ad hoc networks. We start with an overview of some new challenges: how the security aspects (i.e. attacks, mechanisms, and services) differ in an ad hoc network from those in a wired network. Then, we present a comprehensive survey of currently available solutions for mobile ad hoc networks, which answers the question: how are these new challenges tackled? Some recent proposals on key management and secure routing protocols are presented in more details because of their importance and tremendous interest. We hope this chapter could serve as a key to open the door for readers who are interested in grasping and understanding the security related issues in mobile ad hoc networks.

2 Security Challenges

In the computer network and information security context, when there are needs to assess security, to evaluate various network mechanisms, and to choose security products or policies, the following three aspects are usually considered: *security attacks*, *security mechanisms*, and *security services* ([63]). The salient features of mobile ad hoc networks pose new challenges in each of these aspects of security when compared to their wired network counterparts. In this section, we discuss the possible impacts of those ad hoc networking features (e.g. the lack of infrastructure, the node mobility, etc.) on these three aspects of security.

2.1 Security Attacks

A security attack is any action that compromises the security of information illegally or in an unauthorized way. The attacks can be classified into two categories: *passive attacks* and *active attacks*. A passive attack obtains information without proper authorization, while an active attack involves some type of information interruption, modification, or fabrication. Examples of passive attacks are information leakage (via eavesdropping) and traffic analysis (traffic monitoring). Some types of active attacks include

masquerade (impersonating), replay, modification of messages, and denial of service (DoS).

Virtually all kinds of attacks possible in wired networks are possible in a MANET. However, an ad hoc network is generally deployed within a specific area. It is an isolated intranet unless it is connected to the Internet. Such confined communication system actually isolates attackers who are not local to the area. Attackers may exploit the weaknesses of such unique system architecture. It turns out that the mobile ad hoc networking approach does provide some unique vulnerability that an attacker can exploit.

The first vulnerability comes from the wireless channel. The wireless channel is broadcast in nature so it is more susceptible to various forms of attacks such as passive eavesdropping, active signal interference, jamming, and so on. Messages transmitted over the air can be eavesdropped or faked messages can be injected into the network from anywhere without having the physical access to the network components. In addition, the nodes in ad hoc networks are also vulnerable to physical attacks since the nodes usually reside in an open and hostile environment rather than a physically protected place. In a battlefield scenario, the node itself may be captured or compromised.

Traffic monitoring and analysis can be deployed by adversaries to identify the communicating parties and maybe their functionalities. For example, in a tactical MANET without precaution against traffic analysis, an adversary node or nodes may monitor the traffic activities, the nodes with heavy traffic might be the commanding nodes or critical nodes for network connectivity. With this knowledge, the adversaries may be able to take out the important nodes to disable the network. Although many MANET designs have taken the LPI/LPD (low probability of intercept and low probability of detection) into consideration, the LPI/LPD may not be enough against the network penetration, and a passive internal attack is still a possibility.

Another vulnerability made worse by the ad hoc networking approach is attacks on network protocols, both routing protocols and media access control protocols. Restricted by the limited bandwidth, many ad hoc routing protocols are on-demand, which makes them different from the routing protocols used in Internet. All the nodes in an ad hoc network are responsible for routing and forwarding packets. Many ad hoc routing protocols are co-operative in nature and rely on an implicit trust relationship among participating nodes to relay packets. Their co-operative nature makes them more vulnerable to data tampering, impersonation, and denial of service (DoS) attacks ([21, 22, 57]). Moreover, the wireless medium makes it easier for an attacker to inject false information into the network, while the

unpredictable and frequent topological changes make it difficult to distinguish between faked routing information generated by malicious nodes and out-of-date routing information caused by topological changes. Finally, the MAC protocols used in MANET are also co-operative in nature. In either contention-based or reservation-based MAC protocols, all the nodes are supposed to follow the predefined rule to gain the channel access. However, in reality, it is easy for a selfish node to take advantage of this weak point by not following the rules ([36]). For example, it is hard to detect a malicious node who always uses the minimum backoff window size in an MANET using IEEE 802.11 MAC protocol.

2.2 Security Mechanisms

A security mechanism is a mechanism that is designed to provide one or more security services by detecting, preventing, or recovering from one or more security attacks. There is no single mechanism that can provide all the services required in a network and information system. A variety of security mechanisms have been proposed, widely used, and proved effective in the wired Internet. However, certain characteristics of mobile ad hoc networks impede the practical deployment of some security mechanisms that are valid in the wired Internet.

First of all, the lack of a fixed infrastructure or a central concentration point in a mobile ad hoc network makes some conventional security mechanisms based on centralized online servers inapplicable in mobile ad hoc networks. For example, the conventional authentication and encryption schemes using public-key cryptography are based on a centralized trusted certificate authority and intrusion detection systems need a central concentration point to collect audit data. These requirements contradict the infrastructureless and the self-organizing nature of ad hoc networks.

Secondly, nodes in a MANET can move continuously in an unpredictable way. This type of MANET mobility precludes any security solution with a static configuration. In addition, it clouds the distinction between normalcy and anomaly, which makes the detection of the malicious behaviors difficult. For example, as we mentioned before, the mobility causes frequent topological changes, it is very difficult to distinguish between faked routing information and stale routing information. Moreover, as the fundamental security mechanism in virtually all networks, a good cryptographic scheme requires proper management and safe keeping of a small number of cryptographic keys. This design objective is very hard to accomplish in a MANET where nodes can move independently and connectivity is not guaranteed

([24]).

Finally, in MANETs, mobile devices tend to have limited processing power (CPU cycles), limited memory (buffer space), limited transmitting power, limited network bandwidth, and limited battery energy. This severely restricts the practical deployment of more comprehensive or computational intensive, yet more effective, security schemes in MANET environments.

2.3 Security Services

A security service is a service that enhances the security of the network and the information transferred over the network. A number of various security functions have been desirable in a network information system. Based on their objectives, the security services can be categorized into: *Confidentiality, Authentication, Integrity, Non-repudiation, Access Control, and Availability* ([63]). These services are intended to counter one or more attacks, and make use of one or more security mechanisms to achieve their goals. We examine their security properties in a MANET environment next. We present them in a different order, as this often reflects their actual importance in MANETs ([62]).

Availability

Availability requires the network services to be available to authorized parties whenever needed. A variety of denial of service (DoS) attacks can result in the loss of or reduction in availability. Particularly, in a MANET, an adversary could jam the radio frequencies to interfere with signals on physical channels; it could interact with a node in an otherwise legitimate way, but for no other purpose than to deplete others' battery power (it is a more powerful threat than CPU exhaustion for a mobile node ([62]); it could disrupt routing to cripple the network; or it could bring down higher layer services, such as the key management service, a fundamental service for any cryptographic scheme.

Authentication

Authentication ensures that the origin and the destination of a message is correctly identified, with an assurance that the identities between two communicating parties are not falsified. Without authentication, an adversary could masquerade a node, interfere with other nodes' communication, or gain unauthorized transmission and reception. Mobile devices are susceptible to loss, theft, and capture (in a battle field), thus frequent re-authentication

becomes necessary. Again, the absence of an online server poses a fundamental problem in MANETs because the usual authentication mechanisms involve a centralized system entity.

Confidentiality

Confidentiality is the protection of transmitted data from passive attacks, such as eavesdropping. Sensitive information, such as tactical military information or strategic information, requires confidentiality. Leakage of such information to enemies could cause devastating consequences. The other aspect of confidentiality is the protection of traffic flow from analysis. Routing information needs to remain confidential in certain cases, because the source and destination, frequency, length, or other characteristics of the traffic might be helpful for enemies to identify and to locate their targets in a battlefield, or to infer certain tactical information.

Integrity

Integrity ensures that the transmitted information is not illegally modified. Modification includes changing, deleting, creating, delaying or replaying of the transmitted messages. Certain modification could be caused by benign failures, such as the radio propagation impairments. Others are caused by malicious attacks. The integrity of the routing information, particularly the cost metrics, is of great importance in maintaining the proper functioning of the network.

Non-repudiation

A non-repudiation service guarantees that neither the sender nor the receiver of a message is able to deny the transmission. Non-repudiation helps to detect and punish compromised or misbehaving nodes.

Access Control

Access control is the ability to limit and control access to devices and applications via communication links. Each entity attempting to gain access must first be authenticated.

3 Available Security Solutions

In this section, we present proposed security solutions for tackling each of the challenges described in the previous section. In the first five subsections we focus on the security mechanisms and schemes that are proactive and preventive in nature to protect the security of a MANET. In the last two subsections, we summarize the detective and reactive approaches.

3.1 Defending Against Physical Attacks

Mobile devices are susceptible to loss and theft because they are small, light, and easy to carry. In a battle field scenario, they are at risk of being hijacked or captured. It is necessary to protect the physical safety of the mobile devices. The conventional solution to the physical attacks is to implement a security module that is tamper-resistant, i.e. that contains measures to keep data secret and uncorrupted even under physical attacks ([53]). An example is the use of the smart card, which is basically a safe containing a microprocessor and necessary cryptographic information ([1]). The smart card performs all the relevant cryptographic operations and could be inserted into or removed from the device easily. The safe has lid switches and circuitry, which interrupts power to memory, thus key material will be erased when the lid is opened. The advantage of using a removable card is that it allows a user to change devices while keeping his/her own private data. In addition, the sensitive information is protected by the smart card and it can be removed at will. However, when all the information is stored in the smart card, there is still a problem, for example, the device may be stolen with the smart card in it.

An additional protection scheme can be designed to detect whether a device has fallen into the wrong hands. This could be done through user identification and authentication. Some well known techniques include PINs (personal identification numbers), passphrases, and biometics. By requiring user identification periodically and/or for each security-critical transaction, an adversary can be prevented from making a stolen device operational. However, frequent re-authentication is somewhat troublesome and discourages users from activating the security mechanism. Recently a zero-interaction authentication was proposed ([14]), in which a user wears a small authentication token that communicates with the mobile device (such as a laptop) over a short-range wireless link. Whenever the mobile device needs decryption authority (DA), it acquires the DA from the token. The system is automatically protected (re-encrypted) when the user is not around and

the decryption authority cannot be acquired. The system can be restored in seconds once it detects the user's return. The DA is only retained while it is needed. This scheme secures the mobile device from a physical attack while recovering full performance before a returning user resumes to work.

3.2 Enforcing Confidentiality

The wireless channel in MANETs is broadcast in nature. It suffers from poor protection and is particularly vulnerable to passive eavesdropping attacks. This vulnerability is not specific to mobile ad hoc network, but common to all wireless communication networks. In these environments, confidentiality may consists of two aspects: one is to protect the the identity of nodes (either users' identity or the communications entities' functionalities), the other is to protect the transmitted messages from disclosure. The former is particularly important in military applications, in which a node's functionality (e.g. a command node) should be hidden away from non-participating nodes. In what follows in this subsection, we will address both aspects.

3.2.1 Hiding the Nodes' Transmissions

One of the most effective ways to protect the identities of communications entities is to conceal their communications effectively. This can be achieved in the physical layer, where many solutions have been proposed to protect the wireless channel. For example, spread spectrum technologies (e.g. frequency hopping or direct sequence), which either spread the energy in time and/or frequency in a random fashion to make signal capture difficult or spread the energy to a wider spectrum so that transmission power is hidden behind the noise level, can make it difficult to detect or jam signals. The characteristics of LPI/LPD (low probability of intercept/low probability of detection) are highly desirable in military applications. Directional antennas can also be deployed due to the fact that the communication techniques can be designed to spread the signal energy in space. We will not discuss these solutions further as we mainly focus on solutions in high layers.

3.2.2 Securing the Communications Path

One common approach, not unique for MANETs but effective, is to secure the communication path. The basic idea is to encrypt all messages exchanged between communication entities (either point-to-point or end-to-end). However, due to resource (time, frequency and space) constraints, the full version of security schemes used in wired networks may not be effective

in MANETs, instead, light weight versions may have to be developed to fulfill the needs ([17, 52, 67]).

3.2.3 Enhancing Confidentiality via Multipath Routing

As we mentioned earlier, the traditional way of providing a confidentiality service is to apply data encryption/decryption to the information transmitted over the networks. However, the computational burden may pose a serious problem in resource limited environments. Moreover, as we will discuss later in this chapter, key management in MANETs is also problematic due to the infrastructureless architecture.

Another approach enhancing a confidentiality service is to utilize the salient features of MANETs such as the mobility of the network architecture. The fundamental idea comes from the following observation: a messenger who carries the full message from one place to another across hostile ground may reveal the message easier if he/she is captured, while the message will not be fully recovered if multiple messengers are deployed to carry only partial information and go through different routes across the hostile ground. We ([40, 39]) recently developed such a scheme to enhance the confidentiality service on top of any security scheme suitable for the deployed MANET. The proposed scheme, namely, *SPREAD: Secure enhancement Protocol for REliable dAta Delivery*, is based on secret sharing and multipath routing. The basic idea is described as follows. Using a (T, N) secret sharing algorithm, we generate multiple (N) shares of a message (or messages) to be protected, such that from any T or more shares, we can easily recover the message (or messages), while from any $T-1$ or fewer shares, it is computationally impossible to recover the message (messages). Then, using a multipath routing algorithm, we find multiple paths with minimal overlaps (e.g., independent node-disjoint paths) and then we send the N shares over such multiple paths towards the destination. From a network point of view, if a whole message follows a single path to its destination, a hacker can intercept all the necessary information to recover that message at any intermediate node. However, with the SPREAD scheme, the hacker has to compromise a number of nodes on a number of independent paths to obtain at least T different shares. Reduced information interception ratio can be expected from SPREAD.

To better understand the scheme, we give a brief introduction to the threshold secret sharing system, which is also used later for the key management, details can be found in [59]. Suppose that we have a system secret K to be protected, we use it to generate N pieces, S_1, S_2, ..., S_N, called

shares or *shadows*. Each of N participants of the system, P_1, P_2, ..., P_N, hold one share of the secret, respectively. The generation of the secret shares guarantees that any less than T participants cannot learn anything about the system secret K, while with an effective algorithm, any T out of N participants can reconstruct the system secret K. This is called a (T, N) *threshold secret sharing scheme* ([60, 4, 61]). A secret sharing scheme consists of two algorithms. The first is called the *dealer*, which generates and distributes the shares among the participants. The second is called the *combiner*, which collects shares from the participants and recomputes the secret, i.e., it produces the secret K from any T correct shares. A combiner fails to recompute the secret if the number of the correct shares is less than T. For illustration purposes, we take Shamir's Lagrange interpolating polynomial scheme as an example. The dealer obtains the ith participant's share by evaluating a polynomial of degree $(T-1)$:

$$f(x) = (a_0 + a_1 x + a_2 x^2 + \cdots + a_{T-1} x^{T-1}) \bmod p$$

at $x = i$:

$$S_i = f(i)$$

which is given to the participant P_i, where p is a large prime number greater than any of the coefficients and is made available to both the dealer and the combiner, and the coefficient $a_0 = K$ is the secret while other coefficients a_1, a_2, ..., a_{T-1} are all randomly chosen. Then, at a combiner, once T shares have been obtained, the combiner can reconstruct the original blocks by solving a set of linear equations over a finite field. For example, assume that the received T shares are $f(i_1)$, $f(i_2)$, ..., and $f(i_T)$, let

$$A = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{T-1} \end{pmatrix}, B = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ i_1 & i_2 & i_3 & \cdots & i_T \\ i_1^2 & i_2^2 & i_3^2 & \cdots & i_T^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i_1^{T-1} & i_2^{T-1} & i_3^{T-1} & \cdots & i_T^{T-1} \end{pmatrix}, F = \begin{pmatrix} f(i_1) \\ f(i_2) \\ f(i_3) \\ \vdots \\ f(i_T) \end{pmatrix}$$

then, the original secret $a_0 = K$ can be recovered by solving the following linear equations in matrix form

$$B'A = F$$

where B' denotes the matrix transpose of the matrix B. It is known that this equation has a unique solution over the finite field $GF(p)$.

The SPREAD scheme works as follows: if a source node wants to send a message to a destination node securely in a MANET, depending on the security level, the source can use a multipath routing algorithm to find multiple paths from the source to destination with certain properties (for example, disjoint paths in a certain sense), then the source determines a secret sharing scheme, say, a (T, N) threshold scheme, according to the message security level and the availability of the paths with the desirable non-overlapping property. The source will be the dealer in this case, and will choose the a_0 as the message and choose other coefficients randomly on the finite field $GP(p)$, where p is appropriately chosen. Finally, the source node will generate the message shares as discussed earlier and send them over the multiple paths found from the source to the destination. The destination will be a combiner, upon receiving T shares, it is able to recover the original secure message. We observe that SPREAD is built on top of any security scheme, hence it only enhances the confidentiality. Since the multiple paths are chosen in a way that any eavesdropper cannot physically be at two paths, hence he/she will find it hard to collect enough shares for that particular message. Due to the mobility of nodes, even the multiple paths from one node to another will constantly change, which makes possible collusion difficult. Thus, unless a malicious node knows all mobility patterns of the secure communication parties (source and destination), it will be difficult to capture the message flow from the source to the destination when both ends are highly mobile.

A few remarks are in order. First, the SPREAD scheme cannot address confidentiality alone in MANETs, it only statistically enhances this service. For example, a node which can hear all path transmissions can recover the secure message, this is the case if a node near either the source or the destination knows also the spreading codes or hopping pattern used by either the source or destination. The multiple paths in this scenario can be achieved by combining it with either encryption algorithm or directional antenna. Second, if a message is too large compared to the chosen prime number p, the message can be chopped up as we normally do in the transport layer. During this process, some scrambling may be helpful. Third, depending on the number of paths used, SPREAD seems to waste a lot of bandwidth. To save the network bandwidth, in SPREAD all the coefficients a_0, a_1, a_2, ..., a_{T-1} can be assigned messages, so that multiple messages can be delivered simultaneously. Finally, SPREAD can be made adaptive in the sense that the source node could make final decisions whether a message is delivered at a certain time instant according to the security level and the availability of multiple paths. Moreover, the chosen set of multiple paths may be changed

from time to time to avoid any potential capture of those multiple paths by adversaries.

Many variations of SPREAD can be designed according to the above few remarks. Due to the inherited redundancy in SPREAD scheme, we also develop a redundant SPREAD scheme (where $T < N$), which can improve data reliability as the multipath routing has been shown coping with frequent topological changes well, and improving overall throughput ([64, 68]), although it consumes more network bandwidth. In addition, by embedding the cheater detection and identification mechanisms in a secret sharing scheme, the SPREAD scheme is also helpful in identifying malicious or compromised nodes. When combined with the conventional cryptographic schemes, it can reduce computation by using partial share encryption. The major drawback of the SPREAD scheme is that it is limited by the available multiple paths in the MANETs.

3.2.4 Preventing Traffic Analysis

Another threat related to confidentiality is traffic analysis. Traffic analysis is basically the attempt to discover the pattern of traffic between parties. Conventionally, traffic analysis is prevented by link encryption and hiding the real traffic pattern using dummy traffic to form a uniform traffic cloud ([47]). Typically, there are two types of encryption used in a packet-switching network: end-to-end encryption and link encryption. End-to-end encryption is performed at or above the transport layer in the two end systems. When the packets are transmitted over the network, the payload of each transport layer segment, i.e. message content, is encrypted except the header. So the eavesdropper is able to acquire end-to-end flow traffic pattern information such as the source and destination, frequency, duration, and so on. This information can help the adversary to locate the target or to infer the activity or intention of the communication parties. For example, in a military network, a significant change of traffic pattern may imply the deployment of troops, chains of command, level of readiness, etc. In contrast, link encryption is performed at the data link layer by encrypting the payload of each frame. Since multiple end-to-end flows may be multiplexed on each link, an eavesdropper would not be able to distinguish the traffic pattern of end-to-end flows, although the link flow traffic is still observable. Due to the broadcast nature of the wireless channel, the eavesdropper could easily intercept data in an mobile ad hoc network. Thus, link encryption is a more suitable security mechanism for hiding the end-to-end traffic flow information. The wired equivalent privacy (WEP) encryption scheme de-

fined in IEEE 802.11 wireless LAN standard uses link encryption. Moreover, some high level security schemes, which combine multi-path routing, traffic rerouting, and traffic padding techniques, have been proposed to protect end-to-end traffic patterns in a conventional network ([47, 66, 19]).

Recently, a few attempts have been made to prevent traffic analysis specifically for mobile ad hoc networks. The basic strategy to prevent traffic analysis is to create security clouds in a way that each node under a security cloud is identical in terms of traffic generation. In [26] and [27], Jiang, Vaidya and Zhao discussed different methods of constructing a traffic cover mode, e.g., the end-to-end cover mode and the link cover mode. The purpose of the cover mode is to hide the changes of an end-to-end flow traffic pattern as certain tactical information might be inferred from the unusual changes in the traffic pattern. They formulate the construction of the optimal cover mode into an optimization problem and present a solution based on flow rerouting. Their solution is able to find an optimal cover mode for a number of predefined operation modes. However, node mobility and its impact on the calculation of the cover mode are not considered in their solutions. In [28], the authors proposed another approach to hide the source and destination information using the dynamic mix method (DMM). The mix method was originally proposed in [13], which achieves the anonymity of the message delivery via a cryptographic method and "mix" nodes in the network. Due to the dynamic topology changes, when applying the mix method in a MANET, the network performance degrades. The proposed DMM improves the network performance by allowing the communicating nodes to choose mix node dynamically at run time.

3.3 Key Management

Key management is possibly the most critical and complex issue when talking about security in a mobile ad hoc network. The applicability of many other security services, such as confidentiality and authentication, relies on effective and efficient key management. For efficiency reasons, the parties involved in a secure communication usually need to share a common secret key. Public key cryptography has made key distribution easier among those parties in the wired Internet. For example, some public key cryptography based key exchange algorithms, such as the Diffie-Hellman key exchange algorithm ([16]), have been widely adopted to establish the session keys between parties without transmitting the keys themselves in the network. However, the management of the public keys usually involves a centralized trusted control point, called a *certificate authority (CA)*. Such centralized trust control

contradicts the design goal of MANETs, where there is no infrastructure. Some research have been carried out to address the public key management issue in MANETs. Basically, there are two major research directions along this line. One is to retain the certificate authority concept, but distribute its functionality into multiple servers (or trusted nodes) ([42, 30, 75]). In this way, both the availability and the security of the CA can be improved. Another approach is to discard the centralized CA, and instead, create a totally distributed and self-organized key management system ([24]). We will discuss the key management issue in more detail in section 4.

3.4 Authentication

To guarantee secure communication, a node in a MANET has to make sure the party it communicates with is what it claims to be. The process of this verification is authentication, which is done via a challenge process: if the node shares a secret with the other party, the node can issue a challenge message to request the correct answer; or if both parties do not share a secret, a trusted third party (certificate authority) can be used for the verification with the understanding that both parties share secrets with the CA, then the challenge process will be carried out between both parties with the CA. In the traditional Internet, centralized CAs in the fixed infrastructure exist, hence more comprehensive authentication processes can be launched without too many concerns. However, in a MANET, the trust model in the Internet is no longer valid. Mobile devices are powered on and off, move in and out a certain networking domain often, and there is no fixed infrastructure to be trusted, therefore a new trust model has to be developed ([67]). Due to the resource limitations, signaling traffic due to authentication has to be minimized ([37, 38]). The latter issue has been ignored for MANETs in the current literature, while the former issue attracts some attention in conjunction with key management for MANETs ([24, 42, 30, 62, 75]), various distributed trust models have been proposed. Some distributed trust models are developed based on the salient features of MANETs. In [62], the trust model is based on *imprinting*: a duckling recognizes whomever it sees at first sight as its mother and will always obey its mother. Thus, a node will always trust whichever gives its secret and becomes a slave of a mother node, hence authentication can be verified by the mother node. In a distributed public-key trust model ([75]), a selected set of nodes is used as servers (CAs) and collaboratively manages the public keys. Whenever a node needs to have its public key signed, the node has to contact a subset of servers to gain the certificate of its public key, which could be used in the authentication

process when communicating with other nodes. Instead of pre-selecting a set of nodes as the CAs, the authors of [24] suggested a self-organized public-key infrastructure based on a chain of trust. Whenever a node wants to communicate to another party, the trust repositories of both parties will be merged and a search for a trust chain is initiated. Depending on the web of the trust in a MANET, authentication signaling traffic may be huge for both trust models. In [42, 30], Lu and his colleagues proposed a localized distributed trust model: at the very beginning, a system secret key is shared by all nodes in a MANET, each node obtains a share via a threshold sharing scheme with threshold, say, k, i.e., any k shares can recover the original secret. A node can obtain the security services by contacting at least k nearest nodes. In this way, all authentication processes can be localized.

Recently, Weimerskirch and Thonet ([67]) proposed a light-weight authentication model based on the observation that "no model ensures authentication for an ad-hoc network in every environment" and that "depending on the situation users can select the appropriate system". Their proposed trust model targets low-value transactions: it does not make a transaction perfectly secure, but rather makes the attacker's cost to get falsely authenticated higher than the value of the transaction! The trust model is based on human behavior. Assume that a node, say, A wants to authenticate another node, say, B. A can ask some questions (such as a secret or a recent transactions), if the answer is yes, A can trust B. Otherwise, A starts to solicit A's trusted friendly nodes for recommendation, or asks B to provide a list of references for verification. If A's trusted friend node says yes or a reference from B can be authenticated by A and tells A that B is trusted, then A can trust B. This authentication model is useful for MANETs consisting of less powerful processors. More details can be found in [67].

3.5 Protecting Routing Protocols

Protecting routing protocols is another important issue in a mobile ad hoc network. As discussed in Section 2.1, routing in ad hoc networks is more vulnerable than its counterpart in wired networks. Correct routing can be disrupted in many ways or be disabled by denial of service attacks (see section 5.1). Great efforts have been made to protect routing protocols. Several secure routing protocols that aim to protect the correctness of routing protocols have been proposed. Traditional source authentication and message integrity measures have been adopted ([57]). In response to the key management difficulty and the limited resource restrictions, authentication mechanisms without using public key cryptography are also proposed

([21, 22]). Different mechanisms have been proposed for different types of ad hoc routing protocols (table-driven and on-demand, distance vector and source routing). Both preventive schemes and reactive schemes are developed. Due to their importance, section 5 of this chapter will be dedicated to securing routing protocols in more details.

3.6 Handling Node Misbehavior

A less severe attack in MANETs is the node misbehavior caused by selfishness. In an ad hoc network, all basic functions, such as routing and packet forwarding, are collaboratively carried out by all participating nodes. The effectiveness of the MANET design relies heavily on mutual trust and mutual collaboration in sharing the network resources (such as time, frequency, space, code and battery power). It is expected that all participants follow the rules according to the MANET design objective. However, there might exist selfish nodes that do not want to provide services to other nodes for some reasons, e.g. to save their own battery energy ([45]). Or some nodes may want to grasp more bandwidth or demand less delay for their own packets ([36]). Although the selfish node does not perform active attacks, the misbehaviors can cause significant network performance degradation because the MANET depends on the co-operation of all participants to provide services to each other. For example, if $10\% - 40\%$ of the nodes in the network agree to forward packets but fail to do so, the average throughput may degrade by $16\% - 32\%$ ([45]). In this section, we discuss some solutions proposed in the current literature to handle such misbehaviors due to selfishness.

3.6.1 Network Layer Misbehavior

The selfishness in the network layer appears as not forwarding packets for others. Although routing and packet forwarding are two closely related functions in the network layer, the correctness of the routing information does not guarantee the correct forwarding of a packet. Two types of solutions have been proposed to deal with reluctant or even erroneous packet forwarding problems. The first type is reactive in nature. The misbehaving nodes are detected and corresponding reactions are carried out. The other type is to create a incentive mechanism to encourage the cooperation of the nodes.

In [45], Marti et al proposed two techniques that improve throughput in the presence of nodes that agree to forward traffic but fail to do so. A *watchdog* is used to identify misbehaving nodes and a *pathrater* is then designed to help routing protocols avoid these nodes. The watchdog's mechanism is

based on the promiscuous mode of radio interface: the receiver of one node could listen to the transmission of any of its neighbors, regardless of the intended destination of that transmission. Thus, when a node forwards a packet, the node's watchdog verifies that the next node in the path also forwards the packet by overhearing the next node's transmission. If the next node does not forward the packet(s), then it is misbehaving. The pathrater then uses this knowledge to rate the nodes and chooses network paths to avoid the use of misbehaving nodes, thus the packets are most likely to be delivered. This scheme is more for avoiding the problem than facing the problem. If there are too many such misbehaving nodes, the pathrater may not be able to find a feasible path, which will degrade the network performance severely. Some award and punishment strategy may have to be developed and incorporated into this scheme to make it more effective.

Another protocol, using the same type of approach, is the *CONFIDANT (Cooperation Of Nodes: Fairness In Dynamic Ad-hoc NeTworks) protocol* ([7, 8]). The CONFIDANT protocol consists of four major components: the *Monitor*, the *Reputation system*, the *Path manager*, and the *Trust manager*. The monitor performs a similar function as a watchdog, observing the transmission of the next node on the path or observing the routing protocol behaviors. By keeping a copy of a packet while listening to the transmission of the next node, it could also detect any content changes in the frame. If the function that is being monitored provides an acknowledgment message (e.g. the Route Reply message of the DSR protocol), reputation information can be gathered about the nodes that are not within the radio range of the monitoring node. The reputation system rates each node in the network according to the observations about its routing and forwarding behavior, including the first hand observation of neighboring nodes and the trusted second hand observations reported from other nodes. The path manager then performs functions such as path re-ranking according to the reputation of the nodes in the path, deletion of paths containing malicious nodes, action upon receiving a request for a route from a malicious node (e.g. ignore, do not send any reply), and action upon receiving a request for a route containing a malicious node in the source route (e.g. ignore, alert the source). Finally, the trust manager deals with the ALARM messages: it sends out an ALARM message to friends (friendly nodes) when it experiences, observes, or receives a report of malicious behaviors, it also makes a decision on the trustworthiness of an incoming ALARM message.

The CONFIDANT protocol is similar to [45] in the sense that each is a reactive protocol dealing with node selfishness. It improves the detection mechanism by letting nodes learn not only from their own experience,

by observing their neighbors, but also by exchanging experience with their neighbors. However, as we mentioned earlier, the reactive scheme proposed in [45] does not punish malicious nodes that do not cooperate, but rather relieves them of the burden of forwarding for others. The CONFIDANT protocol in contrast isolates the detected selfish nodes so that misbehavior is punished and co-operation is rewarded.

In [9], a different approach was proposed to handle node selfishness. Relying on the observation that sophisticated transactions are usually based on some form of currency, Buttyan and Hubaux proposed a mechanism based on a virtual currency called *"nuglet"*. Nodes pay for the service they receive from a MANET and are paid for service they provide to the other nodes. Two payment models are proposed: *Packet Purse model* and *Packet Trade model*. In the packet purse model, when a node originates a packet, it puts an estimated amount of nuglets in the packet purse attached to that packet. Each node forwarding the packet would then take a certain number of nuglets from that packet purse. If the packet runs out of nuglets in its packet purse, it is dropped. In this way, nodes are given incentive to cooperate by collecting more nuglets for future uses and are discouraged from overloading the network. In a later version of this scheme ([12]), the packet purse is removed from the packet, instead, a *nuglet counter* is implemented in each node. Whenever a packet is originated, the number of nuglets is reduced from the nuglet counter in the source. When a node forwards a packet for others, its nuglet counter is increased. This approach requires a tamper-resistant security module (hardware) in each node to handle the nuglets so that the nuglet counter cannot be changed in an illegal way. It also requires the estimation of the cost from source to destination.

In the packet trade model, the packet does not carry nuglets. Instead, it trades for nuglets with intermediate nodes (neighbors). Each intermediate node buys the packet from the upstream node for nuglets and sells it to the downstream node for more nuglets. The destination actually pays off all the cost. A drawback of this model is that it actually encourages nodes to overload the network since the source does not have to pay.

3.6.2 MAC Layer Misbehavior

Node misbehavior could also happen at the MAC layer. The current wireless LAN MAC protocol IEEE 802.11 (DCF mode) is a contention based protocol. Nodes share the same channel and follow the same exponential backoff procedure when contending for the channel access. The priority that a node wins the contention depends on its backoff value which further heav-

ily depends on the contention window size selected by that node. If a small contention window size is selected, the average deferring time for that node would be short while if a large contention window size is selected, the node might have to back off for a long time period. Therefore, the node misbehavior at the MAC layer typically appears as unfairly obtaining a higher share of the channel or experiencing shorter delay by selecting small backoff value or small contention window size. In [36], Kyasanur and Vaidya proposed a MAC layer node misbehavior detection and correction scheme. The backoff mechanism used in the current 802.11 standard is modified to simplify the misbehavior detection. In the current IEEE 802.11 standard, the backoff value of each station is selected by the station itself, thus other nodes have difficulty in judging if the node follows the backoff procedure or not. In the modified version, the backoff value to be used by a sender for the next transmission to a receiver is selected by that receiver during the current transmission between the sender and receiver. This change allows the receiver to detect the deviation of the actual waiting slots of the sender from the expected waiting slots accurately and promptly. Once a certain number of deviations are detected within a window of certain packets from a node, the node is designated to be misbehaving. Then, a certain amount of backoff time is calculated and added as the penalty to the next backoff time assigned to that node. This correction scheme is actually very weak as it again requires the co-operation of the misbehavior node.

Another approach handling the MAC layer node misbehavior is to develop MAC protocols that are resilient to misbehavior. Game theory seems to be a viable design technique along this direction. According to game theory, the protocol could be designed to reach an optimal operating mode called the *"Nash equilibrium"*. Thus, no node can improve its own interest (such as bandwidth or delay) by changing its own strategy while other nodes' strategies remain fixed. However, the use of game theory in designing self-configuring protocols for wireless mobile ad hoc network has not been well understood. In addition, working at the Nash equilibrium might be far from optimal from the network performance point of view. More work needs to be done along this direction although a few works have demonstrated the applicability of such an approach ([43, 31, 32, 10]).

3.7 Intrusion Detection

In the previous sections, we have focused mainly on the preventive mechanisms to protect network security. As we know, a prevention-only strategy only works if the prevention mechanisms are perfect. Otherwise, someone

may be able to find a way to get around them ([58]). Most of the threats have been the results of bypassing prevention mechanisms. Intrusion detection and response mechanisms in this case provide a second line of defense.

Intrusion detection is based on the assumption that the behavior of the intruder differs from that of a legitimate user in ways that can be quantified ([56]). Generally speaking, there are two approaches detecting an intrusion: misuse (or rule-based) detection and (statistical) anomaly detection. The misuse detection attempts to define improper behavior based on the patterns of well-known attacks. Examples of misuse detection systems include IDIOT ([33]) and STAT ([25]). The misuse detection system can accurately and efficiently detect the known attacks, but it lacks the ability to detect unknown attacks. Statistical anomaly detection attempts to define normal, or expected, behavior. It involves the collection of data relating to the behavior of legitimate users over a period of time. Then, statistical tests are applied to determine with a high level of confidence whether an observed behavior deviates from a legitimate user behavior. An example of anomaly detection system is IDES ([41]). The anomaly detection can be used for a more general purposed protection, including unknown new attacks. However, such systems might cause more false alarms. In practice, both approaches may be combined to be effective against a broader range of attacks.

The intrusion detection systems (IDSs) designed for wired networks do not function well in MANETs. The most important reason is that the intrusion detection systems rely on real time traffic analysis of traces collected at switches, routers, or gateways, while in a MANET environment, there is no such traffic concentration points from which one can collect audit data for the entire network. In addition, MANET mobility, which allows nodes to move freely and independently, complicates the detection because it is harder to distinguish anomaly and normalcy. Other constraints include the availability of only local and partial audit traces, fully distributed requirement, and resource (such as power) limitations ([73]).

In [73, 74], Zhang, Lee and Huang proposed a new "Distributed and Cooperative Intrusion Detection" architecture suitable for mobile ad hoc networks. In this model, an IDS agent runs independently on each node. The IDS agent conceptually consists of six modules: local data collection, local detection engine, local response, global response, cooperative detection engine, and secure communication. The local data collection monitors local activities and collects useful audit data, such as system and user activities within the mobile node, communication activities by this node, and communication activities within the radio range and observed by this node. The local detection engine detects intrusion from these local traces. It includes

both misuse detection and anomaly detection. If a known attack or anomaly with strong evidence is detected, the node can determine independently that the network is under attack. If an anomaly with weak evidence is detected and a broader investigation is required, the node will use the cooperative detection engine to initiate a cooperative global intrusion detection procedure. This procedure includes propagating the intrusion detection state information among nodes and a distributed consensus algorithm to make a decision based on the information received from other nodes. Once an intrusion is determined, the response modules, both the local one and the global one, are responsible for taking actions. The local response module triggers action local to this node while the global one coordinates actions among neighboring nodes. The exact actions taken depend on the type of intrusion, type of network protocols and applications, and the confidence in the evidence. For example, re-authentication between certain nodes or the re-organization of the whole network might be required. Finally, the secure communication module provides a highly secure communication channel among IDS agents.

The proposed intrusion detection architecture is fully distributed and collaborative, and well suitable for MANET environments. In general, it is applicable to all network layers, or in an integrated cross-layer manner. However, intrusion detection heavily depends on the definition of proper behavior or improper behavior that further relies on different applications, protocols, and attacks. As MANET networking is still under active development, we do not yet know what is a "typical" application or scenario. The authors proposed an anomaly detection model for detecting attacks on MANET routing protocols, because routing protocols in MANET are fundamental and well studied. As pointed out by the authors themselves, the more pressing tasks now are to better understand the potential applications for MANETs, and to define realistic benchmarks. Otherwise, it will be very difficult to use intrusion detection in MANETs.

4 Key Management

As we mentioned earlier, key management plays very critical role in addressing any effective security issue. Intensive research has been carried out for MANETs in the last few years. In this section, we present more comprehensive discussions of key management.

4.1 Fundamentals and Overview

Cryptography is the fundamental security technique used in addressing almost all aspects of security. It provides the basis for many security services such as the confidentiality and integrity of messages whenever they are exposed to potential attacks, for example, during the transmission across the networks that are vulnerable to eavesdropping and message tampering. There are two main classes of cryptographic algorithms in general use. The first class is symmetric (or secret-key) cryptography, where the sender and the receiver use the same secret key to encrypt and decrypt the information transmitted between them. The secret key used in this class of algorithms should only be known to the sender and receiver, but not be revealed to anyone else. The second class is asymmetric (or public-key) cryptography, where each participant has a public/private key pair. The public keys are made public to everyone while each participant keeps his/her private key secret. When sending a message, the sender uses the receiver's public key to encrypt the message. The public-key algorithm ensures that only the intended receiver can decrypt the message with his/her private key. The public key algorithm is also used to sign the digital signature for authentication and non-repudiation purpose. When sending a message, the sender signs the message with his/her own private key. The receiver can verify the sender's digital signature with the sender's public key.

Cryptography is widely used in the construction of secure distribution systems. The strength of any cryptographic system depends on proper key management. In symmetric cryptography, if an attacker compromises the keys used in encryption/decryption, then all encrypted messages will be compromised. In asymmetric cryptography, although private keys are not transmitted in the open, the distribution of public keys also causes problem. For example, malicious Tracy could use her public key to replace Alice's public key in a certain key directory to trick Bob to send messages to her. Without proper protection of secret keys in symmetric cryptography or the public keys in asymmetric cryptography, the whole cryptographic system can be easily defeated. Due to the computational complexity of the security schemes, in practice, symmetric cryptography is widely used for bulk data encryption while asymmetric cryptography is used to distribute cryptographic keys as its performance is inadequate for the encryption of bulk data (typically requires 100 to 1000 times as much processing power as symmetric cryptographic algorithms). Moreover, key management for an asymmetric cryptography is also problematic. The management problem here is basically the distribution of public keys. After the public keys

have been distributed or become accessible, well-defined algorithms based on public key cryptography could be applied to sign the digital signature or to distribute session keys for symmetric data encryption, e.g. using key exchange algorithms such as the Diffie-Hellman technique ([16]) or simply encrypting the secret key generated by one party with the public key of the other party. Thus, the problem, key management here, is how to make the public key accessible to all concerned parties without potential abuse of the public key distribution system.

One approach providing public key management service in wired networks or wireless cellular networks is the deployment of a Public Key Infrastructure (PKI). The most important component in a PKI is the *certificate authority* (CA), a trusted entity with its public key known to everyone in the system. The CA is then responsible for issuing, validating and distributing the public keys (in the form of digital certificates) for others. The success of a PKI depends on the availability and security of the CA. However, a PKI does need a central control point, which everybody trusts, this is possible only when certain fixed infrastructure exists. Thus, the difficulty in applying a PKI in a MANET is that such a central control point is not available. Even when available, it cannot be well protected in MANETs and would become the most vulnerable point in the system. To deal with this problem, a distributed and cooperative CA was proposed ([75]). The basic idea is that, by secret sharing, the service and the trust of a CA is distributed into a set of nodes (also called servers), which can be trusted to a certain extent, a certain number of nodes cooperatively perform the functions of a CA, so that the service remains available and correct even if a small number of such nodes or servers become unavailable or compromised. Substantial work has been carried out in this direction recently ([75, 76, 30, 70]). We will elaborate this approach in Section 4.2.

Another approach is to deploy distributed key management without a CA. Distributed key management, as used in PGP (Pretty Good Privacy) ([63, 59]), solves the "no CA" problem with introducers. The introducers are other users of the system who sign their friends' public keys. The basic idea is as follows. If A knows B, then A signs B's key and gives B a copy of the signature. When B meets a stranger C, B presents his key with the signature of A. If C also knows and trusts A, C has reason to trust B. By this means, the users sign each other's key and over time, each one will collect many introducers. Then, they have high probability of verifying each other's key by one of their introducers. A similar approach as in PGP was adopted in [24], where the authors proposed a self-organized public key management system for fully self-organized mobile ad hoc networks. We will discuss this

approach in Section 4.3.

It is worth noting that providing a security service always starts from a certain *prior context*, consisting of the well defined name space, each node's security requirement, and some prior trust relationship ([2]). For example, the centralized PKI system with CA is based on the prior trust between each node and CA while the distributed key management, as used in PEP, is based on the prior trust between friends (e.g. authentic nodes in MANETs). These prior trusts must be done out-of-band, through possibly non-cryptographic ways, such as physical contact or other secure channel. Security techniques can help only in transforming and transferring the trust assumptions in the prior context. They cannot create trust.

More key management mechanisms based on different types of prior context have been proposed. In [2], Asokan and Ginzboorg proposed a solution for key agreement for a MANET conferencing scenario. It is assumed that there is no public key infrastructure or physically secure communication channels. However, all the participants are able to choose and share a fresh password (e.g. by writing it on a blackboard). Starting from this weak shared password, the authors proposed a password-based authenticated key exchange to establish a stronger shared key among participants. However, this proposal assumes that the participants in the conference do not change during the session. If the members are dynamic, the session keys needs to be updated when the composition of the group changes. In another paper [46], Montenegro and Castelluccia proposed using a crypto-based identifier (CBID) to simplify the key management and provide an auto-configurable foundation for nodes to engage in verifiable information exchanges with each other. The idea is to have an implicit cryptographic binding between a node's identifier and its public key (or certificate). Ideally, in identity-based public key cryptography ([59]), the binding should be done by generating public/private key pair based on the node's ID. However, the mathematics of these sorts of schemes turns out to be infuriatingly complicated to make secure ([59]). Otherwise, a trusted authority is required to establish the private keys among the users. In [46], the authors proposed the binding by generating the node's ID from self-generated public key. However, this actually poses another challenge, that of managing the name (ID) space of the MANET, which is not desirable.

For a critical survey on key establishment protocols in a mobile communications network with infrastructure, readers are referred to [6].

4.2 Distributed and Cooperative Certificate Authority

PKI is the most popular public key management system. In the PKI system, the public keys are distributed in the form of public-key certificates. A public-key certificate is a node's public key signed by a trusted entity, which also contains information about that node. The trusted entity is a certificate authority (CA). Prior trust between the CA and each node is assumed (the initial authentication can be done by non-cryptographic means such as physical contact). Then, it is up to the CA to manage the public-key certificate of each node with proper authentication. The major certification services provided by a CA include certificate issue, renewal, revocation, and certificate directory service ([70]).

CA is a centralized control mechanism. Its success depends on the availability and the security of the CA. In a MANET, due to the frequent change of topology , it is not easy to guarantee that a single node is always accessible. Moreover, setting up a single node as a CA would provide the adversaries a single most vulnerable point. Once the CA is compromised, the whole system will be subverted. Based on these considerations, a distributed and cooperative CA model was proposed recently by Zhou and Haas ([75]).

In this proposed model, the distribution of trust is achieved by using threshold cryptography. As we introduced in section 3.2.3, an $(n, t+1)$ (denoted as (T, N) in Section 3.2.3) threshold cryptography scheme allows n parties to share the ability to perform a cryptographic operation, so that any $t+1$ or more parties can perform this operation jointly, though it is infeasible for t or fewer parties to do so, even by a possible collusion. With an $(n, t+1)$ configuration ($n \geq 3t+1$), there are n special nodes, called *servers*, collectively perform the functions of a CA. The system private key K is divided into n shares $(s_1, s_2, \ldots, s_n)$, each share is given to each server. Each server has its corresponding private/public key pair and stores the public keys of all the nodes in the network. In particular, each server knows the public keys of all other servers.

For the service to sign a certificate, each server generates a partial signature for the certificate using its private key share and submits the partial signatures to a combiner. With $t+1$ correct partial signature, the combiner is able to compute the signature for the certificate. However, with t or fewer partial signatures, it is computationally impossible to compute the signature of the certificate. Thus, to subvert the CA, the enemy has to compromise at least $t+1$ servers. It is worth noting that the signing process here is not the generation of an commom digital signature. In order to maintain the property of the secret sharing, when a secret share is used in signing par-

tial certificate, it is treated as an exponent in RSA algorithm. More details about the signing process was described in [30].

To tackle the mobile adversaries attack, in which an adversary compromises one server and then moves to the next victim, the authors proposed to use *share refreshing*. Share refreshing allows the servers to compute new shares from the old shares in collaboration without disclosing the system private key to any server. Share refreshing is based on the following homomorphic property. If $(s_{11}, s_{21}, \ldots, s_{n1})$ are the $(n, t+1)$ secret shares of K_1 and $(s_{12}, s_{22}, \ldots, s_{n2})$ are the $(n, t+1)$ secret shares of K_2, then $(s_{11}+s_{12}, s_{21}+s_{22}, \ldots, s_{n1}+s_{n2})$, where " + " is the addition operation on a finite field, are the $(n, t+1)$ secret shares of key $K_1 + K_2$. If K_2 is 0, then a new $(n, t+1)$ secret shares of K_1 are obtained. The new shares are independent of the old ones, so the mobile adversary has to compromise at least $t+1$ servers during one refreshing period. A share refreshing process is also proposed, and it requires only $t+1$ subshares to generate the new shares. In addition, the configuration of the system could be changed from $(n, t+1)$ to $(n', t'+1)$ adaptively, if necessary.

The proposed distributed CA model has been implemented in COCA (Cornell Online CA) ([76]), although it is not dedicated to MANET. A similar approach has been adopted in [30] and [70]. In [30], a self-initialization protocol is proposed to handle dynamic node membership (i.e. joins and leaves) and secret share updates. The nice feature of this approach is that providing the security services can be localized and the trust can be made more robust (see also [42]).

4.3 Self-organized Public Key Management

As we mentioned before, PGP, a widely used Email security system, uses a distributed approach for key management. There is no centralized key certification authority. Instead, PGP supports a "web of trust". Every user generates and distributes his/her own public key. Users sign each other's public keys, creating an interconnected community of PGP users. Similar to that used in PGP, an alternative key management system, *the self-organizing public key management system*, was proposed by Hubaux and his colleagues in their TERMINODES project for fully self-organized mobile ad hoc networks ([5, 18, 24]).

In this proposed system, each user maintains a local certificate repository that contains a limited number of certificates selected by the user according to some algorithm. When user u wants to obtain or verify the public key of user v, the user u will merge the local certificate repositories of both user u

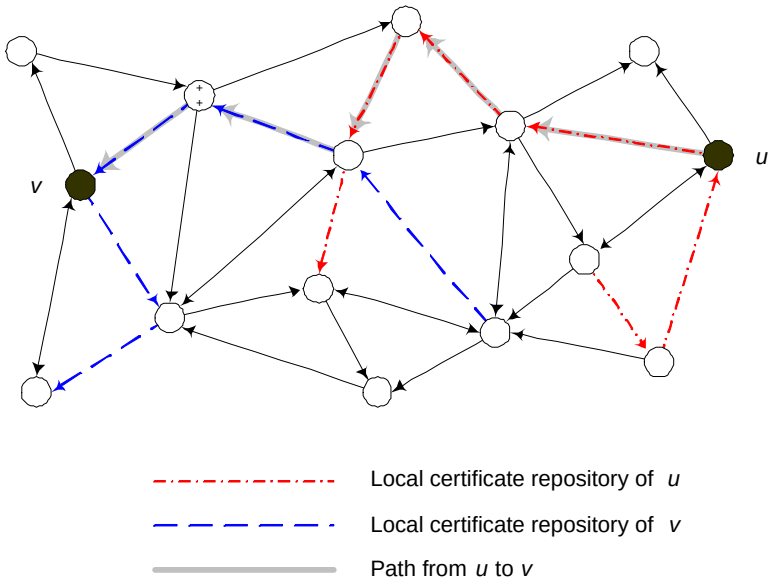


Figure 1: Merging subgraphes of node u and v for chain of trust between them ([24])

and user v, and tries to find an appropriate certificate chain from u to v in the merged repository. The authors use a directed graph $G(V, E)$, called a *trust graph*, to represent the trust relationship between users. As shown in Figure 1, if user u issues a public-key certificate to user v, there would be a directed edge from vertex u to vertex v in the graph. Then, a directed path from vertex u to vertex v indicates a certificate chain from user u to user v.

The local certificate repository maintained at each node is represented as a subgraph of that node. The success of this approach depends on the construction of the local certificate repositories and the characteristics of the trust graph. The size of each local repository should be kept small compared with the total number of users in the system for scalability. On the other hand, when combining their repositories, any pair of legal users should be able to find a certificate chain between them with high probability. Two algorithms are proposed for users to build their local certificate repositories: *shortcut hunter algorithm* and *star shortcut hunter algorithm*. Here, a *shortcut* is defined as an edge, such that once it is removed, the shortest undirected path between the nodes previously connected by that edge becomes strictly larger than two. The basic idea of the shortcut hunter al-

gorithm is to build a subgraph that consists of a single out-bound path and a single in-bound path. Path selection is similar for both paths. It starts from the node itself. In each round, it selects a node among neighbors of the last selected vertex. The selection criterion is to choose the one that has the highest number of shortcuts. The star shortcut hunter algorithm modified the pure shortcut hunter algorithm a little bit. Instead of finding a single out-bound and a single in-bound path, it builds a subgraph that consists of several vertex disjoint out-bound paths and several vertex disjoint in-bound paths.

This is a fully distributed and also scalable approach to the key management system. However, it only provides probabilistic guarantees. In addition, this approach assumes that trust is transitive, which is often not the case in practice. In order to alleviate this problem, the authors proposed using multiple certificate paths and using authentication. Similar to PGP, the weakest link of this whole system is key revocation, which was briefly addressed in their work.

5 Secure Routing Protocols

Routing is of paramount importance in a mobile ad hoc network because traditional Internet routing protocols are no longer effective due to the frequent topological changes. A great effort has been made within the Internet Engineering Task Force (IETF) mobile ad hoc networking (MANET) working group in order to develop a routing framework for IP-based protocols in MANETs. A number of routing protocols have been proposed and widely evaluated. The proposed routing protocols can be generally divided into two main categories: *table-driven (proactive)* and *on-demand (reactive)* ([55]). Similar to the routing protocols used in wired networks, table-driven routing protocols [50] attempt to maintain consistent, up-to-date route information from each node to every other node, regardless of the need for such routes. They respond to changes in topology by propagating updates throughout the network. An on-demand routing protocol ([29, 49]) differs from this in that it attempts to discover a route to a destination only when it has a packet to forward to the destination. Discovered routes are maintained by a route maintenance procedure until either the destination becomes inaccessible along every path from the source or until the route is no longer desired. In addition, some ad hoc routing protocols are hybrid with a combination of the table-driven and on-demand mechanisms ([20]). These routing protocols are generally designed with the objective of keeping up with the

frequent and unpredictable changes in topology. More sophisticated protocols consider routing scalability, route stability, energy efficiency, and so on. Most of the designs assume a trusted environment and the co-operation of each participating node. Relatively little research has been done in a more realistic setting where an adversary may attempt to disrupt operation.

5.1 Vulnerabilities Analysis of Mobile Ad Hoc Routing Protocols

As mentioned in Section 2.1, without incorporating protection mechanisms, ad hoc network routing protocols are more vulnerable to many types of attacks. Generally speaking, the attacks to a routing protocol can be first divided into two classes: *passive attacks* and *active attacks*. The passive attacks on routing protocols are eavesdropping only, but not endangering message transmissions. It is mainly a threat to the message confidentiality, and does not affect the proper functioning of the routing protocols. The passive attacks on routing protocols can be readily protected in the same way as for data traffic. So we do not discuss it further here. The active attacks, however, can be further classified into two classes: *external attacks* and *internal attacks*. External attacks are from outsiders, such attacks are limited when encryption and source authentication are in place. A special case of external attacks is the wormwhole attack, which will be discussed shortly. Internal attacks, coming from internal nodes (e.g. compromised nodes), are more severe attacks because an internal node has all the necessary credentials for authentication and so on. A single node or multiple nodes could launch an attack individually without collusion, or multiple nodes could also launch attacks as a team with shared information and coordinated collaboration. Most of the secure routing protocols proposed in the current literature can only prevent individual attacks, either from internal or from external sources, but have limited capability to handle collusion attacks.

With this classification, we are now ready to identify and classify the possible attacks to routing protocols ([57, 22]). We discuss the protection against the attacks briefly.

5.1.1 Modification of Routing Information

A general attack to disrupt routing function is to modify the routing information. Here, by modification, we assume that the source and destination nodes are trusted while the attack is performed by any intermediate node

while processing the transit routing information. For example, for a table-driven protocol, a malicious node could send out false routing updates, or for an on-demand protocol, a malicious node could alter the information contained in the route request or route reply in a route discovery process. The altered routing information could cause legitimate traffic to be redirected to a black hole or an inefficient detour (a black hole is where all the packets are dropped except routing packets), or a routing loop is formed, or even a network partition may be caused ([22]). Any information field in a routing message could be exploited by an attacker. For example, the *destination_sequence_number* is widely used in a mobile ad hoc network routing protocol to indicate the freshness of the routes. A malicious node could simply modify the *destination_sequence_number* to make other routing information invalid while making itself the freshest route, or it could modify the hop counts to claim the shortest path to any destination, or in protocols such as DSR, any intermediate node could simply modify the replied routes, or a malicious node could also illegally modify its IP and/or MAC address to impersonate another node (spoofing).

This type of modification attack is basically an attack by illegally modifying the content in the routing messages, carried out by either an external attacker or an internal attacker. This type of attack can be prevented by source authentication and message integrity services. The illegal modification of the stationary fields such as the node address or *destination_sequence_number* can be protected by message integrity measures such as a message authentication code (MAC). While protection of the integrity of the routing metric (e.g. hop count) for a hop-by-hop routing protocol and the complete route (node lists) information contained in some on-demand routing protocols is a little different, as such information is allowed to change at each hop. Much effort has been made along this line to guarantee the correctness of the routing operation and routing information. Several secure routing protocols have been proposed that are robust to these attacks when performed by individual attackers. We will discuss the proposed secure routing protocols in Section 5.2.

5.1.2 Fabrication of Routing Information

Another form of attack is to fabricate false topology information. By fabrication, we mean that the false routing information is initiated by a malicious node. This happens particularly for route maintenance processes. For example, in any routing protocols such as DSDV ([50]), AODV ([49]) or DSR ([29]), a malicious node can fabricate routing updates or route error mes-

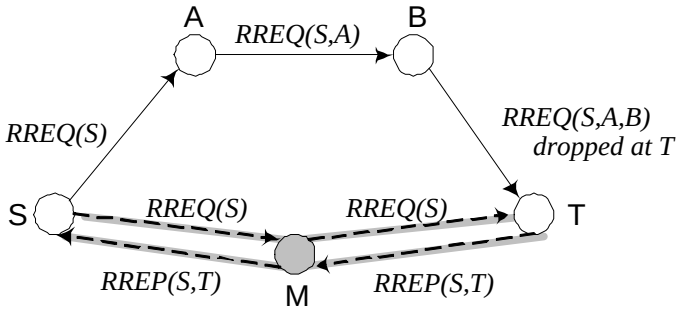
sages to claim the inaccessibility of another node. Source authentication can limit this type of attacks to the extent that the malicious node could only claim the inaccessibility to its own neighbors. Thus, this attack actually causes less significant damage to the network, as the effect of this attack is the isolation of the malicious node itself. This type of attack cannot be prevented for an internal attacker. However, non-repudiation protection can be applied to facilitate the detection of such attackers ([57]).

5.1.3 Replay

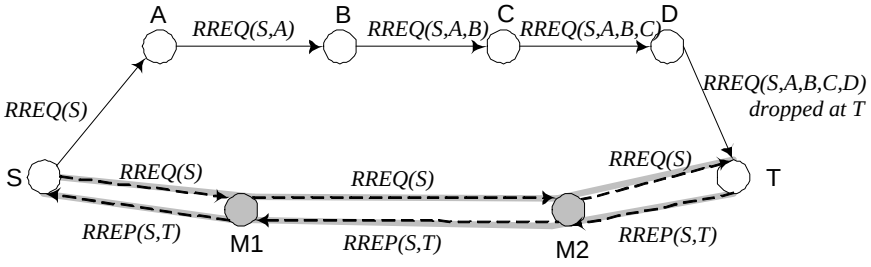
A replay attack captures a data unit passively and then retransmits it to produce an unauthorized effect. One of the replay attacks in MANETs is to create a *wormhole* ([23]). In a wormhole attack, the attacker records packets, particularly routing control packets, at one location in the network, tunnels them to another location, and then retransmits them into the network there. It could be either an external attack or an internal attack. This attack is dangerous because the source may fail to find routes or find routes that actually do not exist. An individual wormhole attack is shown in Figure 2(a) where a malicious node M simply replay the routing request and reply message between node S and T without showing itself. Thus, an actually non-exist path S-T would be replied to S. Figure 2(b) shows a collusive wormhole attack, where two malicious nodes M_1 and M_2 collaborate to make the tunneled packets arrive sooner than other packets traveling over a normal multihop route. In order to minimize the delay introduced by the wormhole, the attack(s) can replay the packet bit by bit without storing the whole packet, or can use some long-distance wireless link or even wired link to tunnel the packet ([23]).

A subtler replay type of attacks is *tunneling* ([57]). A tunneling attack is where two or more nodes may collaborate to encapsulate and exchange messages between them along existing data routes. This attack requires the collusion of internal attackers. An example of the tunneling attack is shown in Figure 3, where malicious node M_1 and M_2 tunnels RREQ and RREP messages using the data path M_1-B-C-M_2. However, the source would wrongly consider route S-M_1-M_2-D to be a shorter path.

The external wormhole attack can be partially prevented at the physical layer, e.g, by using a secret modulation method, RF watermarking, or tamper-resistant hardware ([23]). However, if the attackers manage to capture and replicate the waveform, this approach is likely to fail. Prevention of such an attack by a software-only approach is difficult, if not impossible, as the attack takes advantage of the operation of the routing protocols. How-



(a). Individual attack



(b). Collaborative attack (collusion)

Figure 2: Wormhole attack

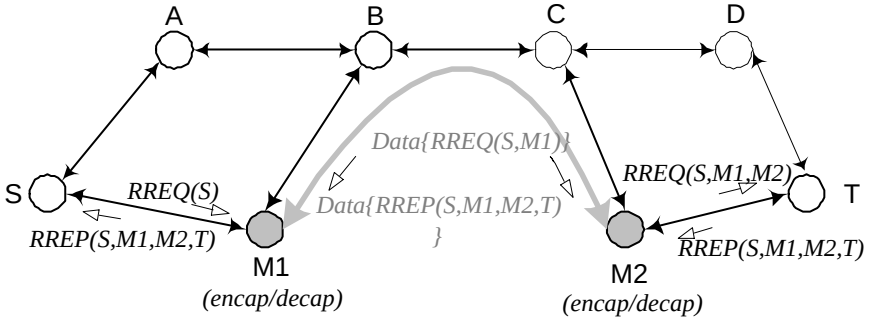


Figure 3: Tunneling attack

ever, detection of such an attack is possible by using some unalterable and independent physical metric such as time delay or geographical location. We will discuss a mechanism dealing with this problem in the next section.

5.1.4 Denial of Service (DOS)

The DoS attack on a routing protocol could take many forms, but all result in the unavailability of the routing service. The consequences of many of the above attacks are denial of service. Besides disrupting routing, an attacker could also launch a DoS attack by excessively consuming network resources. For example, a malicious node could initiate excessively unnecessary route discovery processes or it could inject extra packets into network with the sole purpose of wasting other nodes' energy when processing and forwarding the packets. This type of attacks, if performed by an internal node, is difficult to prevent. However, as pointed out in [22], an attack is considered a DoS attack only if the ratio between the total work performed by nodes in the network and the work performed by the attacker is on the order of the number of nodes in the network. e.g., a single packet sent by the attacker results in a packet flood throughout the network. Thus the DoS attack can be efficiently limited by preventing an attacker from inserting routing loops, or enforcing a maximum route length a packet can traverse. Moreover, an active detection scheme could also be applied to identify such attacks ([65]), as we will discuss in the next subsection.

5.2 Protection of Routing Protocols

Much of the effort in providing security in mobile ad hoc networks has been spent on protecting routing protocols. In general, a secure routing protocol should implement some kind of authentication and integrity schemes so that the correctness of the routing information, particularly, the node identity, the *destination_sequence_number*, and the cost metric, can be protected. Several secure routing protocols have been proposed along this line, each with different authentication schemes and for different types of routing protocols.

5.2.1 Hop-by-hop Authentication

For distance vector routing protocols such as DSDV and AODV, due to their operational features, the correctness of the routing information has to be provided on a hop-by-hop basis. Moreover, the correct accumulation of the routing metric has to be guaranteed, i.e. an internal malicious node

should not be able to reduce the routing metric from itself to any other destination. In [21], Hu, Johnson and Perrig proposed a *Secure Efficient Distance Vector (SEAD)* routing protocol to provide authentic routing information for the proactive DSDV protocols. The authors suggested several options for neighbor authentication, such as computing an Message Authentication Code (MAC) for each neighbor with each routing update, to protect fields such as source identity and destination sequence number. Particularly, they proposed using a one-way hash chain to protect the metric field. The one-way hash chain is a series of data generated from a one-way hash function, which is easy to compute in one way while infeasible to do in the reverse. Thus, in a one-way hash chain, any following datum can be calculated from a previous datum, however, a previous datum cannot be derived from a following datum. In the proposed scheme, the one-way hash chain is generated from one direction, but used in the reverse direction. A hash value corresponding to both the sequence number and the metric is used to authenticate each entry in the routing update. Due to the one-way nature of the hash chain, any intermediate node is not able to modify a route to some destination with a lower metric. This scheme does not use the computationally intensive asymmetric cryptographic operations. However, it does require some mechanism for a node to distribute an authentic element from its generated hash chain initially and periodically when necessary, as the authentication of a value in the hash chain is based on an earlier authentic value. A similar approach has been used in S-AODV ([72]) as an extension of the AODV routing protocol. A one-way hash chain is used for each route discovery to secure the hop count information while public key cryptography (digital signature) is used to authenticate the sources as well as other non-mutable fields of the messages. In [57], Sanzgiri et al implemented the hop-by-hop authentication for the route control messages of AODV and DSR. The protocol uses public key cryptography (digital signature) to guarantee message authentication, integrity and non-repudiation. It provides end-to-end authentication by disabling the option in a route discovery process to reply from intermediate node. The assumption of the existence of a centralized CA in the proposed protocol limits the applicable scenarios to classroom or conference type of managed-open environments.

5.2.2 End-to-end Authentication

For source routing based protocols such as DSR, authentication can be done on an end-to-end basis because the end nodes have knowledge of the complete route. However, the integrity of the routes (e.g., the complete and

correct node list in the route) needs to be carefully protected in this case. In [48], Papadimitratos and Haas proposed a secure routing protocol (SRP) to provide end-to-end authentication for source routing based protocols. The correctness of the replied routes is protected by adding message authentication codes (MACs) at the source to the route requests, and at the destination to the route reply messages. The scheme assumes a prior security association (SA) between source and destination while the existence of SAs with any of the intermediate nodes is not necessary. In [22], Hu, Perrig and Johnson proposed another secure routing protocol (Ariadne) based on DSR. The authentication also relies on the message authentication code and is end-to-end in nature. The destination node authenticates the route request messages. In addition, a per-hop hashing technique is presented to verify that no node is missing from the node list in the Route Request message and that all the nodes listed in the Route Request are legitimate nodes. They introduced the use of a broadcast authentication scheme, called *TESLA* ([51]), which requires loose time synchronization, to provide the authenticity of the routing information for the DSR protocol. Because the authentication is end-to-end in nature, both schemes require the route replies from the destination while disabling the route replies from intermediate nodes. This might cause performance degradation of routing protocols ([44]).

5.2.3 Reactive Approach

The afore-mentioned basic mechanisms are generally proactive and preventive methods to secure routing protocols. The objective is to secure the correctness of the exchanged routing information as well as the routing operation. More mechanisms are proposed to enhance routing security using reactive approaches. One motivation of using reactive mechanism is to protect the attacks that are difficult to prevent. In [23], Hu, Perrig and Johnson proposed a mechanism, namely *packet leashes*, to defend against the wormhole attacks (it also applies to tunneling attacks). The basic idea is to use additional timing and/or location information so that a receiver can determine if the packet has traveled a route that is not realistic for the specific network technology used. Two types of packet leashes are considered: geographical leashes and temporal leashes. The temporal leashes rely on extremely precise time synchronization and extremely precise timestamps in each packet, while the geographical leashes use location information and loose time synchronization.

The network layer plays two major functions: routing and packet forwarding. They are closely related: the packet forwarding is based on the

routing decision. A reactive approach can monitor the behaviors of both the routing function and the forwarding function. So another motivation for a reactive approach is to provide unified network layer protection. In [69], Yang, Meng and Lu proposed such a unified network-layer security protocol that protects both routing and forwarding functions in the context of AODV. In their design, each authentic node is assigned a temporary token to participate in the network. The routing and packet forwarding behavior of each node is constantly monitored by his/her neighbors. Once the token is expired, the neighbors collaboratively renew the node's token based on a threshold secret sharing scheme ([42]). Only when the node's behavior is approved by a certain number (threshold) of neighbors, is the node able to renew its token. Otherwise, it will be isolated from the network. This scheme takes a fully localized design without completely trusting any individual node. In [65], Venkatraman and Agrawal proposed a model that combines both external attack prevention and internal attack detection for AODV. The message authentication code is used to prevent external attacks, while an internal detection and response module is designed to identify the misbehaving nodes, and then isolate them. Their detection mechanism is similar to a rule-based intrusion detection system and both the routing and packet forwarding behaviors are monitored. The AODV protocol is modified so that two-hop routing information is maintained at each node for each route. Each node constantly monitors and analyzes the behavior of its neighbors based on certain predefined attack patterns. The fabrication attacks are monitored in their scheme. Work adopting a similar approach can also be found in [3], where a protocol consisting of both the preventive mechanism and the fault detection mechanism was proposed. However, the objective of the detection is to detect Byzantine behavior, which is defined as any action by an authenticated node that results in disruption or degradation of the routing service. The detection is result based, without considering the reason that causes the fault. An adaptive probing technique was proposed, which is based on acknowledgments of the data packets.

In [71], Yi, Naldurg and Kravets presented a *security-aware routing (SAR)* protocol. Nodes are assumed to have different trust levels and the SAR is designed to find the path consisting of only nodes with a required trust level. Nodes with the same level of trust share the same secret keys. Nodes with different trust levels are not able to process the routing control messages. SAR is actually not a secure routing protocol in the strict sense, but an approach more to quality of service (QoS) routing in which the security is treated as a QoS parameter.

6 Conclusions

Security is an important issue in mobile ad hoc networks. The current research and development in this area is still in its infancy. In this chapter, we examine the security aspects and present a comprehensive survey on the state-of-the-art development in the security related issues in mobile ad hoc networks. We discuss many security issues, mainly from a network perspective, and present the recently proposed solutions. Two major research focuses, key management and secure routing protocols, are presented in more details. Due to the space limitations, only the novel and fundamental idea of each approach is presented here. The readers are referred to each individual paper for the mechanism or protocol details as well as the performance evaluations.

References

[1] R. Anderson, M. Kuhn, "Tamper resistance - a cautionary note," *Proceedings of the Second USENIX Workshop on Electric Commence*, pp. 1-11, Oakland, CA, November 1996.

[2] N. Asokan and P. Ginzboorg, "Key agreement in ad-hoc networks," *Computer Communications*, **23**:1627-1637, 2000.

[3] B. Awerhuch, D. Holmer, C. Nita-Totaru and H. Rubens, "An on-demand secure routing protocol resilient to Byzantine failure," *ACM WiSe'02*, September 2002.

[4] G.R. Blakley, "Safeguarding Cryptographic Keys," *Proc. AFIPS 1979 National Computer Conference*, vol.48, pp.313-317, New York, June 1979.

[5] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.-P. Hubaux and J.-Y. Le Boudec, "Self-organization in mobile ad hoc networks: the approach of terminodes," *IEEE Communication Magazine*, vol. 39, issue 6, pp. 166-174, June 2001.

[6] C. Boyd and A. Mathuria, "Key establishment protocols for secure mobile communications: a critical survey," *Computer Communications*, 23:575-587, 2000.

[7] S. Buchegger and J.-Y. Le Boudec, "Performance analysis of the CONFIDENT protocol," *Proceedings of the 3rd ACM International Sym-*

posium on Mobile Ad Hoc Networking and Computing (MobiHOC'02), June 2002.

[8] S. Buchegger and J.-Y. Le Boudec, "Nodes bearing grudges: towards routing security, fairness, and robustness in mobile ad hoc networks," *Proceedings of 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing (EUROMICRO-PDP)*, pp. 403-410, 2002.

[9] L. Buttyan and J.-P. Hubaux, "Enforcing service availability in mobile ad hoc networks," *Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'00)*, pp. 87-96, 2000.

[10] L. Buttyan and J.-P. Hubaux, "Rational exchange - a formal model based on game theory," *Proceedings of the 2nd International Workshop on Electrical Commerce (WELCOM)*, November 2001

[11] L. Buttyan and J-P. Hubaux, "Report on a working session on security in wireless ad hoc networks," *ACM Mobile and Computing and Communication Review*, vol.6, no.4, 2002.

[12] L. Buttyan and J.-P. Hubaux, "Stimulating co-operation in self-organizing mobile ad hoc networks," *ACM/Kluwer Mobile Networks and Applications (MONET)*, to appear 2002.

[13] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84-88, February 1981

[14] M. D. Corner and B. D. Noble, "Zero-interaction authentication," *the 8th ACM International Conference on Mobile Computing and Networking (MobiCom'02)*, pp. 1-11, Atlanta, GA, September 2002.

[15] H. Deng, W. Li and D. P. Agrawal, "Routing security in wireless ad hoc networks," *IEEE Communications Magazine*, pp. 70-75, October 2002.

[16] W. Diffie and M. Hellman, "New direction in cryptography," *IEEE Transaction on Information Theory*, pp.644-654, 1976.

[17] A. Fox and S. Gribble, "Security on the move: indirect authentication using Kerberos," *IEEE/ACM MobiCom'96*, New York, 1996.

[18] T. Gross, J.-P. Hubaux, J.-Y. Le Boudec and M. Vetterli, "Toward self-organized mobile ad hoc networks: the terminodes project," *IEEE Communication Magazine*, vol. 39, issue 1, pp. 118-124, January 2001.

[19] Y. Guan, X. Fu, D. Xuan, P. U. Shenoy, R. Bettati and W. Zhao, "NetCamo: Camouflaging network traffic for QoS-guaranteed mission critical applications," *IEEE Transactions on Systems, MAN, and Cybernetics - Part A: Systems and Humans*, pp. 253-265, vol.31, no.4, July 2001.

[20] Z. J. Haas, M. R. Pearlman and P. Samar, "The zone routing protocol (ZRP) for ad hoc networks," IETF Internet draft, draft-ietf-manet-zone-zrp-04.txt, July 2002

[21] Y.-C. Hu, D. B. Johnson and A. Perrig, "SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks," *The 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02)*, pp. 3-13, June 2002.

[22] Y.-C. Hu, A. Perrig and D. B. Johnson, "Ariadne : a secure on-demand routing protocol for ad hoc networks," *the 8th ACM International Conference on Mobile Computing and Networking (MobiCom 2002)*, pp. 12-23, September 2002.

[23] Y.-C. Hu, A. Perrig and D. B. Johnson, "Packet leashes: A defense against wormhole attacks in wireless ad hoc networks," Technical Report TR01-384, Department of Computer Science, Rice University, Dec 2001.

[24] J-P. Hubaux, L. Buttyan and S. Capkun, "The quest for security in mobile ad hoc networks," *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'01)*, 2001.

[25] K. Ilgun, R. A. Kemmerer and P. A. Porras, "State transition analysis: a rule-based intrusion detection approach," *IEEE Transaction on Software Engineering*, 21(3):181-199, March 1995.

[26] S. Jiang, N. Vaidya and W. Zhao, "Routing in packet radio networks to prevent traffic analysis," *Proceedings of the IEEE Information Assurance and Security Workshop*, pp. 96-102, West Point, NY, July 2000.

[27] S. Jiang, N. H. Vaidya and W. Zhao, "A dynamic mix method for wireless ad hoc networks," *IEEE Military Communications Conference (Milcom'01)*, pp. 873-877, McLean, VA, October 2001.

[28] S. Jiang, N. Vaidya and W. Zhao, "Preventing traffic analysis in packet radio networks," *DARPA Information Survivability Conference & Exposition II (DISCEX'01)*, vol.2, pp. 163-158, 2001

[29] D. B. Johnson, D. A. Maltz, Y-C. Hu and J. G. Jetcheva, "The dynamic source routing protocol for mobile ad hoc networks," IETF Internet Draft, draft-ietf-manet-dsr-07.txt, February 2002.

[30] J. Kong, P. Zerfos, H. Luo, S. Lu and L. Zhang, "Providing robust and ubiquitous security support for manet," *Proceedings of the 9th IEEE International Conference on Network Protocols(ICNP)*, pp. 251 -260, 2001.

[31] J. Konorski, "Protection of fairness for multimedia traffic streams in a non-cooperative wireless LAN setting," PROMS 2001, vol. 2213 of LNCS, Springer

[32] J. Konorski, "Multiple access in ad-hoc wireless LANs with non-cooperative stations," NETWORKING 2002, vol. 2345 of LNCS, Springer

[33] S. Kumar and E. H. Spafford, "A software architecture to support misuse intrusion detection," *Proceedings of the 18th National Information Security Conference*, 1995.

[34] Y. Kwon, Y. Fang, and H. Latchman, "A novel medium access control protocol for wireless local area networks," *IEEE INFOCOM'2003*, San Francisco, California, March/April 2003.

[35] Y. Kwon, Y. Fang, and H. Latchman, "Fast collision resolution (FCR) MAC algorithm for wireless local area networks," *IEEE Globecom'2002*, Taipei, Taiwan, November 2002.

[36] P. Kyasanur and N. H. Vaidya, "Detection and handling of MAC layer misbehavior in wireless networks," Technical Report, CSL, UIUC, Aug 2002

[37] Y.-B. Lin and Y.K. Chen, "Reducing authentication signaling traffic in third generation mobile network," To appear in *IEEE Transactions on Wireless Communications*.

[38] Y.-B. Lin, S. Mohan, N. Sollenberger and H. Sherry, "Adaptive algorithms for reducing PCS network authentication traffic," *IEEE Transactions on Vehicular Technology*, 46(3):588-596, 1997.

[39] W. Lou and Y. Fang, "SPREAD: Improving network security by multipath routing in ad hoc networks," Technical Report, Dept. of Electrical and Computer Engineering, University of Florida, 2002.

[40] W. Lou and Y. Fang, "A multipath routing approach for secure data delivery," *IEEE Military Communications Conference (Milcom'01)*, vol. 2, pp. 1467 -1473, McLean, VA, October 2001.

[41] T. Lunt, A. Tamaru, F. Gilham, R. Jagannathan, P. Neumann, H. Javitz, A. Valdes and T. Garvey, "A real-time intrusion detection expert system (IDES) - final technical report", Technical Report, Computer Science Laboratory, SRI International, Menlo Park, CA, February 1992.

[42] H. Luo and S. Lu, "Ubiquitous and robust authentication services for ad hoc wireless networks," Technical Report UCLA-CSD-TR-200030, Department of Computer Science, UCLA, 2000.

[43] A. B. MacKenzie and S. B. Wicker, "Game theory and the design of self-configuring, adaptive wireless networks," *IEEE Communication Magazine*, pp. 126-131, November 2001.

[44] D. A. Maltz, J. Broch, J. Jetcheva and D. B. Johnson, "The effects of on-demand behavior in routing protocols for multihop wireless ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1439-1453, August 1999.

[45] S. Marti, T. Giuli, K. Lai and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," *the 6th annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00)*, pp.255-265, Boston, MA, USA, August 2000.

[46] G. Montenegro and C. Castelluccia, "Statistically unique and cryptographically verifiable (SUCV) identifiers and addresses," *Proceedings of the 9th annual network and distributed system security symposium (NDSS)*, February 2002.

[47] R. E. Newman-Wolfe and B. R. Venkatraman, "High level prevention of traffic analysis," *Proceedings of the 7th Annual Computer Security and Applications Conference*, pp. 102-109, December 1991.

[48] P. Papadimitratos and Z. J. Haas, "Secure routing for mobile ad hoc networks," *Proceedings of the SCS communication networks and distributed systems modeling and simulation conference (CNDS 2002)*, San Antonio, TX, January 2002.

[49] C. E. Perkins, E. M. Belding-Royer and S. R. Das, "Ad hoc on-demand distance vector (AODV) routing," IETF Internet draft, draft-ietf-manet-aodv-12.txt, November 2002.

[50] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *Computer Communication Review*, October 1994, pp. 234-244.

[51] A. Perrig, R. Canetti, D. Tygar and D. Song, "Efficient authentication and signature of multicast streams over lossy channels," *Proceedings of the IEEE Symposium on Security and Privacy*, May 2000.

[52] A. Perrig, R. Szewczyk, J.D. Tygar, V. Wen, and D.E. Culler, "SPINS: security protocols for sensor networks," *ACM Wireless Networks*, 8(5), 521-534, September 2002.

[53] A. Pfitzmann, B. Pfitzmann and M. Waidner, "Trusting mobile user devices and security modules," *IEEE Computer*, February 1997.

[54] B. Radosavljevic and B. Jajek, "Hiding traffic flow in communication networks," *IEEE Military Communications Conference (Milcom'92)*, October 1992.

[55] E. M. Royer and C-K Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications*, pp.46-55, April 1999.

[56] D. Samfat and R. Molva, "IDAMN: an intrusion detection architecture for mobile networks," *IEEE JSAC*, vol.15, no.7, pp.1373-1380, 1997.

[57] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields and E. M. Belding-Royer, "A secure routing protocol for ad hoc networks," *the 10th IEEE International conference on network protocols (ICNP)*, November 2002.

[58] B. Schneier, *Secrets and Lies: Digital Security in a Network World*, John Wiley & Sons, 1st edition, 2000.

[59] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, John Wiley & Sons, 2nd edition, 1996.

[60] A. Shamir, "How to Share a Secret," *Communications of the ACM*, 22(11):612-613, November 1979.

[61] G. J. Simmons, "An Introduction to Shared Secret and/or Shared Control Schemes and The Application," *Contemporary Cryptology: The Science of Information Integrity*, IEEE Press, pp.441-497, 1992.

[62] F. Stajano and R. Anderson, "The resurrencting duckling: security issues for ad-hoc wireless networks," *Proceedings of the 7th International Workshop on Security Protocols, Lecture Notes in Computer Science 1796*, pp. 172-182, Springer-Verlag, Berlin, 1999.

[63] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 2nd edition, Prentice Hall, 1999.

[64] A. Tsirigos and Z.J. Haas, "Multipath routing in the presence of frequent topological changes," *IEEE Communication Magazine*, pp. 132-138, November 2001.

[65] L. Venkatraman and D.P. Agrawal, "Strategies for enhancing routing security in protocols for mobile ad hoc networks," *Journal of Parallel and Distributed Computing*, to appear.

[66] B. R. Venkatraman and R. E. Newman-Wolfe, "Transmission schedules to prevent traffic analysis," *Proceedings of the 9th Annual Computer Security and Applications Conference*, pp. 108-115, December 1993.

[67] A. Weimerskirch and G. Thonet, "A distributed light-weight authentication model for ad-hoc networks," *Lecture Notes in Computer Science*, No. 2288, pp.341-354, 2002.

[68] K. Wu and J. Harms, "Performance study of a multipath routing method for wireless mobile ad hoc networks," *the 9th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 99-107, 2001.

[69] H. Yang, X. Meng and S. Lu, "Self-organized network-layer security in mobile ad hoc networks," *ACM WiSe'02*, September 2002.

[70] S. Yi and R. Kravets, "Key management for heterogeneous ad hoc wireless networks," Technical Report No. UIUCDCS-R-2002-2290, UIUC, July 2002.

[71] S. Yi, P. Naldurg and R. Kravets, "Security-aware ad-hoc routing for wireless networks," Report No. UIUCDCS-R-2001-2241, Department of Computer Science, UIUC, Aug 2001.

[72] M. G. Zapata, "Secure ad hoc on-demand distance vector (SAODV) routing," Internet draft, draft-guarrero-manet-saodv-00.txt, Aug 2002

[73] Y. Zhang and W. Lee, "Intrusion detection in wireless ad hoc networks," *Proceedings of the 6th ACM International Conference on Mobile Computing and Networking (MobiCom'00)*, August 2000.

[74] Y. Zhang, W. Lee and Y. Huang, "Intrusion detection techniques for mobile wireless networks," *ACM/Kluwer Mobile Networks and Applications (MONET)*, to appear.

[75] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network Magazine*, vol. 13, no. 6, pp. 24-30, November/December 1999.

[76] L. Zhou, F. B. Schneider and R. V. Renesse, "COCA: a secure distributed on-line certification authority," *ACM Transactions on Computer Systems*, to appear.