Network Security
Scott Huang, David MacCallum, and Ding Zhu Du(Eds.) pp. – - –

Routing Security in Ad Hoc Wireless Networks [1]

Mohammad O. Pervaiz, Mihaela Cardei, and Jie Wu
Department of Computer Science and Engineering
Florida Atlantic University, Boca Raton, FL 33431
E-mail: {mpervaiz@, mihaela@cse., jie@cse.}fau.edu

Contents

[1]This works was supported by the DoD Defense-wide RDTE grant on Secure Telecommunication Networks.

1 Introduction to Ad Hoc Wireless Networks

Wireless networks provide rapid, untethered access to information and computing, eliminating the barriers of distance, time, and location for many applications ranging from collaborative, distributed mobile computing to disaster recovery (such as fire, flood, earthquake), law enforcement (crowd control, search and rescue) and military communications (command, control, surveillance, and reconnaissance). An ad hoc network is a collection of wireless mobile hosts forming a temporary network without the aid of any established infrastructure or centralized administration [12].

In ad hoc wireless networks, every device has the role of router and actively participates in data forwarding. Communication between two nodes can be performed directly if the destination is within the sender's transmission range, or through intermediate nodes acting as routers (multi-hop transmission) if the destination is outside sender's transmission range.

Some of the characteristics which differentiate ad hoc wireless networks from other networks are:

1. **Dynamic Network Topology**. This is triggered by node mobility, nodes leaving or joining the network, node inoperability due to the

lack of power resources, etc. Nonetheless, the network connectivity should be maintained in order to allow applications and services to operate undisrupted.

2. **Fluctuating Link Capacity**. The effects of high bit error rate are more profound in wireless communication. More than one end-to-end path can use a given link in ad hoc wireless networks, and if the link were to break, could disrupt several sessions during period of high bit transmission rate.

3. **Distributed Operations** The protocols and algorithms designed for an ad hoc wireless network should be distributed in order to accommodate a dynamic topology and an infrastructureless architecture.

4. **Limited Energy Resources** Wireless devices are battery powered, therefore there is a limited time they can operate without changing or replenish their energy resources. Designing energy efficient mechanisms are thus an important feature in designing algorithms and protocols. Mechanisms used to reduce energy consumption include (a) having nodes enter sleep state when they cannot send or receive data, (b) choose routing paths that minimize energy consumption, (c) selectively use nodes based on their energy status, (d) construct communication and data delivery structures that minimize energy consumption, and (e) reduce networking overhead.

Designing communication protocols in the ad hoc wireless networks is challenging because of the limited wireless transmission range, broadcast nature of the wireless medium (hidden terminal and exposed terminal problems [15]), node mobility, limited power resources, and limited physical security. Advantages of using an ad hoc wireless networks include easy and speedy deployment, robustness (no infrastructure required), adaptive and self-organizing network.

In this chapter we are concerned with security of routing protocols in ad hoc wireless networks. Routing is an important operation, providing the communication protocol for data delivery between wireless devices. Assuring a secure routing protocol is a challenging task since ad hoc wireless networks are highly vulnerable to security attacks due to their unique characteristics. Traditional routing protocols designs do not address security, and are based on a mutual trust relationship between nodes.

The rest of this chapter is organized as follows. We continue with an overview of the routing protocols in ad hoc wireless networks in section

2. Security services and challenges in an ad hoc network environment are presented in section 3. We continue with a classification and description of the main attacks on routing in section 4, followed by a description of the state-of-the-art security mechanisms for routing protocols in section 5. Our article ends in section 6 with conclusions.

2 Overview of Routing Protocols in Ad Hoc Wireless Networks

Routing is an important operation, being the foundation of data exchanging between wireless devices. Each wireless node acts as a router and participate in the routing protocol. Routing relies therefore on an implicit trust relationship among participating devices. Main routing responsibilities are exchanging the routing information, finding a feasible path between source and destination based on various metrics, and path maintenance.

The major requirements [15] of a routing protocol are (1) minimum route acquisition delay, (2) quick route reconfiguration in the case of path breaks, (3) loop-free routing, (3) distributed routing protocol, (4) low control overhead, (5) scalability with network size, (6) QoS support as demanded by the application, (7) support of time-sensitive traffic, and (8) security and privacy.

There are a number of challenges [15] triggered by the unique characteristics of ad hoc wireless networks. Node mobility affects network topology and may incur packet lost, path disconnection, network partition and difficulty in resource allocation. Wireless nodes are in general resource constrained, in terms of battery power, memory and computing power. Wireless channel has a high bit error rate (10^{-5} to 10^{-3}) compared with wired counterparts (10^{-12} to 10^{-9}). Wireless channel is shared by the nodes in the same broadcast area, thus the link bandwidth available per node is limited, and varies with the number of nodes present in that area. The design of routing protocols should take these factors into consideration.

Based on the routing information update mechanism, routing protocols in ad hoc wireless networks can be classified as proactive (or table-driven) protocols, reactive (or on-demand) protocols, and hybrid routing protocols. In the next three subsections we present important features of each category and short descriptions of several representative routing protocols.

2.1 Proactive Routing Protocols

In proactive routing protocols, nodes exchange routing information periodically in order to maintain consistent and accurate routing information. When a node has to transmit data to a destination, the path can be computed rapidly based on the updated information available in the routing table. The disadvantage of using a proactive protocol is high overhead needed to maintain an up to date routing information. In ad hoc wireless networks, node mobility triggers a dynamic topology that might require a large number of routing updates. This has a negative impact on resource constrained wireless devices, bandwidth utilization, and throughput.

The protocols in this category are typically extensions of the wired network routing protocols. Examples include Destination Sequence Distance Vector (DSDV) [19], Wireless Routing Protocol (WRP) [14], Optimized Links State Routing (OLSR) [3], etc.

Next we present the main features of DSDV [19]. A security enhancement mechanism (SEAD [7]) for DSDV will be detailed later in section 5.1. Similar with other distance vector protocols, DSDV finds shortest paths between nodes using a distributed version of the Bellman-Ford algorithm. Each node maintains a routing table, with an entry for each possible destination in the network. For each entry, the following fields are maintained: the destination address, next hop on the shortest path to that destination, shortest known distance to this destination, and a destination sequence number that is created by the destination itself. To maintain an updated view of the network topology, each node sends periodically to each of its neighbors its routing table information. Based on the routing information received from its neighbors, each node updates its routing table to reflect current status of the network.

Sequence numbers play an important role in DSDV and are used for preventing loop formation. Each entry in the routing table has a sequence number. This is the most recent sequence number known for that destination, and is included in the periodic routing updates. If a node receives an update with a smaller sequence number, then that update is ignored. A newly advertised path is adopted if it has a greater sequence number, or if it has the same sequence number but a lower metric.

Besides the periodic updates, there are triggered updates, issued when important routing updates should be transmitted. When a broken link is detected, the node creates a routing update with next odd sequence number and metric value of infinity. Routing update messages can be full dump,

when information for all destination is sent, or incremental when only information changed from the last full dump is sent.

Main advantage of using DSDV is that routes to all destinations are always available, without requiring a route discovery process. Main disadvantage of DSDV is high overhead due to the periodic routing updates.

2.2 Reactive Routing Protocols

In the reactive routing protocols, a route discovery mechanism is initiated only when a node does not know a path to a destination it wants to communicate with. In the case of mobile ad hoc network, reactive routing protocols have been demonstrated to perform better with significantly lower overheads than proactive routing protocols since they are able to react quickly to the many changes that may occur in node connectivity, and yet are able to reduce (or eliminate) routing overhead in periods or areas of the network in which changes are less frequent.

A reactive routing protocol has two main operations, route discovery (usually broadcasting using a form of controlled flooding) and route maintenance. Various reactive protocols have been proposed in literature such as Ad Hoc On-demand Distance Vector (AODV) [20], Dynamic Source Routing (DSR) [12], Temporally Ordered Routing Algorithm (TORA) [18], etc. We present next the main features of DSR and AODV. Security supporting mechanisms for these protocols are presented later in section 5.

DSR [12] is a source routing protocol, and thus has the property that each data packet carries the source-destination path in its header. Using this information, intermediate nodes can determine who is the next hop this packet should be forwarded to. Each node maintains a routing cache that contains routing information that the node learned from routing information forwarded or overheard. Every entry has an expiration time after which the entry is deleted in order to avoid stale information.

DSR performs route discovery by having the sender broadcasts by flooding a *RouteRequest* packet. Each *RouteRequest* contains a sequence number generated by the source node, in order to prevent loop formation and to avoid multiple retransmissions by a node of the same *RouteRequest* packet. An intermediate node checks the sequence number, and appends its own identifier and forwards the *RouteRequest* only if this message is not a duplicate. The receiver, upon receiving the *RouteRequest*, sends back a *RouteReply* packet along the reverse route recorded in *RouteRequest*. Upon receiving the *RouteReply*, the sender starts sending data to the receiver.

As part of the route maintenance, if a node detects a failure (e.g. broken link), it sends a *RouteError* message to the source. All intermediate nodes hearing the *RouteError* update their routing cache and all routes that contain this hop are truncated. If the source does not have an alternative path to the destination, it has to re-initiate the path discovery mechanism.

DSR has several optimization techniques. First, it allows intermediate nodes that know a path to the destination to reply to the *RouteRequest* message instead of forwarding the request. This speeds up the route discovery. Secondly, path discovery can use an expanding ring search mechanism when sending the *RouteRequest* messages. This is especially useful for close destinations, thus avoiding broadcasting in the whole network.

Advantages of DSR include (1) route maintenance apply only to active routes, (2) route caching can speed up and reduce overhead of route discovery, and (3) a single route discovery might yield more routes to the destination when intermediate nodes reply from local caches. Disadvantages of DSR are: (1) adding the source-destination path in each packet incurs overhead, especially for long paths and small data, (2) the flooding used in route discovery is unreliable, redundant, may introduce collisions, contentions, and (3) intermediate nodes might send *RouteReply* from stale routing caches, thus polluting other caches as well.

AODV [20] implements the same main operations as DSR. It discovers a path to a destination using a *RouteRequest* and *RouteReply* sequence, and performs route maintenance for link failures by propagating a *RouteError* message to the source. AODV tries to improve on DSR by maintaining routing tables at the nodes, such that data packets do not contain the source-destination path. Each node maintains a routing table for each destination of interest, including the following fields: destination, next hop, number of hops, destination sequence number, and expiration time.

When a source node broadcasts a *RouteRequest* to discover a path to a destination, intermediate nodes that forward the message set up a reverse path, pointing toward the node from which the request was received. In this way *RouteReply* travels along the reverse paths set-up when *RouteRequest* was forwarded, without carrying the full path in the header. When *RouteReply* travels along the reverse path, each node sets up forward links that will be used later to forward data packets between the source and destination. When a source node sends a *RouteRequest*, it assigns a higher sequence number for that destination. Intermediate nodes are allowed to reply with *RouteReply* only if they know a recent path to the destination (with the same or higher sequence number). The reverse and forward paths are purged from

the routing tables if they are not used within a specific time interval.

The advantages of AODV can be summarized as follows: (1) paths are not included and carried in the packet headers, (2) nodes maintain routing tables with entries only for the active routes (if not used for specific time interval they are purged), and (3) AODV uses a destination sequence number mechanism to limit the chances of an intermediate node replying with stale information to a *RouteRequest* packet.

2.3 Hybrid Routing Protocols

Some ad hoc network routing protocols are hybrid of proactive and reactive mechanisms. Examples of hybrid routing protocols are Zone Routing Protocol (ZRP) [6], Core Extraction Distributed Ad Hoc Routing Protocol (CEDAR) [23], etc.

ZRP [6] is a hybrid of proactive and reactive routing protocols. The network is divided in zones, where every zone is a r-hop neighborhood of a node. The intra-zone routing protocol is a proactive routing protocol, while the inter-zone routing protocol is a reactive routing protocol. By varying r, we can control the routing update control traffic. When a node wants to transmit data to a destination within the same zone, then this is done directly using the proactive routing protocol, and the information already available in routing tables.

If the destination is in another zone, then the source node bordercasts the *RouteRequest* (e.g. this message is forwarded by the border routers) until it reaches the destination zone. The border node of the destination zone sends then back a *RouteReply* message. Any node forwarding the *RouteRequest* appends its address to it. This information is used when sending *RouteReply* back to the source.

If a broken link is detected, the path reconstruction can be done locally, and then a path update is sent to the source, or can be done globally, by having the source re-initiate the path discovery.

ZRP efficiently explores the features of proactive and reactive protocols. It reduces the control overhead by maintaining the proactive protocols within zones, and reduces the flooding drawbacks by deploying the reactive protocol and bordercast mechanism only between the zones. Particular attention should be considered when selecting the zone radius r, since this can significantly impact the routing performance.

2.4 Broadcasting in Ah Hoc Wireless Networks

Broadcasting refers to the operation of sending a message to all other hosts in the network. Broadcasting is used for the route discovery in reactive routing protocols. In a mobile environment, broadcasting is expected to be used more frequently since nodes mobility might trigger path disconnecting and thus route discovery is invoked as part of the path maintenance procedure.

Broadcasting operation has the following characteristics [26]: (1) the broadcast is spontaneous, that means that each node can start broadcasting at any time, and (2) broadcasting is unreliable. No acknowledgment packet is sent for example in IEEE 802.11 by a node upon receiving a broadcast message.

One straightforward method used to implement broadcasting is through a form of controlled flooding. In this method, each node retransmits a broadcast message when it receives it first time. Transmitting a broadcast through flooding in a *CSMA/CA* network triggers a numbers of issues, commonly referred to as the *broadcast storm problem* [26]:

1. **Redundant rebroadcast**. A node resends a broadcast message even if all its neighbors have already received the message from some other neighbors.

2. **Contention**. The neighbors of a transmitting node receive the message at approximately the same time, and when re-sending the message, they contend for the wireless communication medium.

3. **Collision**. Collisions are more likely to occur because of the lack of back-off mechanism and the lack of RTS/CTS dialogue. Such an example is when more neighbors retransmit at the same time a message recently received.

The work of [26] proposes several schemes to alleviate the *broadcast storm problem*, by limiting the cases when a node rebroadcasts a message: (1) probabilistic scheme, when each node rebroadcasts a message with a specific probability, (2) counter-based scheme, when a node retransmits a message if it was received less than a threshold number of times over a fixed interval, (3) distance-based scheme, when a message is resent only if it is received from neighbors farther away than a specific threshold distance, and (4) location-based scheme, when a node retransmits a message only if the additional area covered is larger than a specific threshold area.

The work of [27] proposes several local and deterministic schemes where a subset of nodes, called forward nodes, is selected locally while ensuring broadcast coverage. In one scheme, each node decides its own forwarding status, whereas in another scheme, the status of each node is determined by neighbors jointly.

In section 5.11, we discuss few mechanisms proposed recently in literature to secure the broadcast operation.

3 Security Services and Challenges in Ad Hoc Wireless Networks

In order to assure a reliable data transfer over the communication networks and to protect the system resources, a number of security services are required. Based on their objectives, the security services are classified in five categories [24]: availability, confidentiality, authentication, integrity and nonrepudiation.

- **Availability**: Availability implies that the requested services (e.g. bandwidth and connectivity) are available in a timely manner even though there is a potential problem in the system. Availability of a network can be tempered for example by dropping off packets and by resource depletion attacks.

- **Confidentiality**: Confidentiality ensures that classified information in the network is never disclosed to unauthorized entities. Confidentiality can be achieved by using different encryption techniques so that only the legitimate communicating nodes can analyze and understand the transmission. The content disclosure attack and location disclosure attack reveals the contents of the message being transmitted and physical information about a particular node respectively.

- **Authenticity**: Authenticity is a network service to determine a user's identity. Without authentication, an attacker can impersonate any node, and in this way, one by one node, it can gain control over the entire network.

- **Integrity**: Integrity guarantees that information passed on between nodes has not been tempered in the transmission. Data can be altered both intentionally and accidentally (for example through hardware glitches, or in case of ad hoc wireless connections through interference).

- **Non-repudiation**: Non-repudiation ensures that the information originator can not deny having sent the information. This service is useful for detection and isolation of compromised nodes in the network. Many authentication and secure routing algorithms implemented in ad hoc networks rely on trust-based concepts. The fact that a message can be attributed to a specific node helps making these algorithms more secure.

Designing a secure ad hoc wireless networks communication is a challenging task due to (1) insecure wireless communication links, (2) absence of a fixed infrastructure, (3) resource constraints (e.g. battery power, bandwidth, memory, CPU processing capacity), and (4) node mobility that triggers a dynamic network topology.

The majority of traditional routing protocols design fail to provide security. The main requirements [15] of a secure routing protocol are: (1) detection of malicious nodes; such nodes should be avoided in the routing process, (2) guarantee of correct route discovery, (3) confidentiality of network topology; if an attacker learns the network topology, he can attack the bottleneck nodes, detected by studying the traffic patters. This will result in disturbing the routing process and DoS, and (4) stability against attacks; the routing protocol must be able to resume the normal operation within a finite amount of time after an attack.

4 Security Attacks on Routing Protocols in Ad Hoc Wireless Networks

Providing a secure system can be achieved by preventing attacks or by detecting them and providing a mechanism to recover for those attacks. Attacks on ad hoc wireless networks can be classified as active and passive attacks, depending on whether the normal operation of the network is disrupted or not.

1. **Passive Attack**: In passive attacks, an intruder snoops the data exchanged without altering it. The attacker does not actively initiate malicious actions to cheat other hosts. The goal of the attacker is to obtain information that is being transmitted, thus violating the message confidentiality. Since the activity of the network is not disrupted, these attackers are difficult to detect. Powerful encryption mechanism

can alleviate these attackers by making difficult to read overheard packets.

2. **Active Attack:** In active attacks, an attacker actively participates in disrupting the normal operation of the network services. A malicious host can create an active attack by modifying packets or by introducing false information in the ad hoc network. It confuses routing procedures and degrades network performance. Active attacks can be divided into internal and external attacks:

   External Attacks are carried by nodes that are not legitimate part of the network. Such attacks can be defended by using encryption, firewalls and source authentication. In external attacks, it is possible to disrupt the communication of an organization from the parking lot in front of the company office.

   Internal Attacks are from compromised nodes that were once legitimate part of the network. Since the adversaries are already part of the ad hoc wireless network as authorized nodes, they are much more severe and difficult to detect when compared to external attacks.

A large number of attacks have been identified in literature that affect the routing in ad hoc wireless networks. Solutions and mechanism that defense against various attacks are presented later in section 5. Next, we classify routing attacks into five categories: attacks using impersonation, modification, fabrication, replay, and denial of service (DoS).

4.1 Attacks using Impersonation

In impersonation attacks, an intruder assumes the identity and privileges of another node in order to consume its resources or to disturb normal network operation. An attacker node achieves impersonation by misrepresenting its identity. This can be done by changing its own IP or MAC address to that of some other legitimate node. Some strong authentication procedures can be used to stop attacks by impersonation.

Man-in-the-Middle Attack
In this attack, a malicious node reads and possibly modifies the messages between two parties. The attacker can impersonate the receiver with respect to the sender, and the sender with respect to the receiver, without having either of them realize that they have been attacked.

Sybil Attack
In the Sybil attack [16], an attacker pretends to have multiple identities. A malicious node can behaves as if it were a larger number of nodes either by impersonating other nodes or simply by claiming false identities. Sybil attacks are classified into three categories: direct/indirect communication, fabricated/stolen identity, and simultaneity. In the direct communication, Sybil nodes communicate directly with legitimate nodes, whereas in the indirect communication messages sent to Sybil nodes are routed through malicious nodes. An attacker can fabricate a new identity or it can simply steal it after destroying or temporarily disabling the impersonated node. All Sybil identities can participate simultaneously in the network or they may be cycled through.

4.2 Attacks using Modification

This attack disrupts the routing function by having the attacker illegally modifying the content of the messages. Examples of such attacks include redirection by changing the route sequence number and redirection with modified hop count that can trigger the black hole attack. Some other modification based attacks are presented next.

Misrouting Attack
In the misrouting attack, a non-legitimate node sends data packet to the wrong destination. This type of attack is carried out by modifying the final destination address of the data packet or by forwarding a data packet to the wrong next hop in the route to the destination.

Detour Attack
In this type of attack, the attacker adds a number of virtual nodes in to a route during the route discovery phase. As a consequence, the traffic is diverted to other routes that appear to be shorter and might contain malicious nodes which could create other attacks. The attacking node can save energy in a detour attack because it does not have to forward packets to that destination itself. This attack is specific to source routing protocols.

Blackmail Attack
Blackmail attack causes false identification of a good node as malicious node. In ad hoc wireless networks, nodes usually keep information of perceived malicious nodes in a *blacklist*. An attacker may blackmail a good node and tell other nodes in the network to add that node to their blacklists as well, thus avoiding the victim node in future routes.

4.3 Attacks using Fabrication

In fabrication attacks, an intruder generates false routing messages, such as routing updates and route error messages, in order to disturb network operation or to consume other node resources. A number of fabrication messages are presented next.

Resource Consumption Attack
In this attack, a malicious node deliberately tries to consume the resources (e.g. battery power, bandwidth, etc.) of other nodes in the network. The attack can be in the form of unnecessary route requests, route discovery, control messages, or by sending stale information. For example, in *routing table overflow* attack, a malicious node advertises routes to non-existent nodes, thus causing routing table overflow. By using *packet replication* attack, an adversary consumes bandwidth and battery power of other nodes.

Routing Table Poisoning
In this attack, a malicious node sends false routing updates, resulting in sub-optimal routing, network congestion, or network partition.

Rushing Attack
A malicious node in rushing attack attempts to tamper *RouteRequest* packets, modifying the node list, and hurrying its packet to the next node. Since in on demand routing protocol only one *RouteRequest* packet is forwarded, if the route requests forwarded by the attacker are first to reach target (destination), then any route found by the route discovery mechanism will include a path through the attacker.

Black Hole
In this type of attack, a malicious node advertise itself as having the shortest path to all nodes in the network (e.g. the attacker claims that it is a level-one node). The attacker can cause DoS by dropping all the received packets. Alternately, the attacker can monitor and analyze the traffic to find activity patterns of each node. Sometimes the black hole becomes the first step of a man-in-the-middle attack.

Gray Hole
Under this attack, an attacker drops all data packets but it lets control messages to route through it. This selective dropping makes gray hole attacks much more difficult to detect then blackhole attack.

4.4 Replay Attacks

In the replay attack, an attacker retransmits data to produce an unauthorized effect. Examples of replay attacks are wormhole attack and tunneling attack.

Wormhole Attack
In the wormhole attack [11], two compromised nodes can communicate with each other by a private network connection. The attacker can create a vertex cut of nodes in the network by recording a packet at one location in network, tunneling the packet to another location, and replaying it there. The attacker does not require key material as it only needs two transceivers and one high quality out-of-band channel. The wormhole can drop packets or it can selectively forward packets to avoid detection. It is particularly dangerous against different network routing protocols in which the nodes consider themselves neighbor after hearing a packet transmission directly from some node.

Tunneling Attack
In a tunneling attack [22], two or more nodes collaborate and exchange encapsulated messages along existing data routes. For example, if a *RouteRequest* packet is encapsulated and sent between two attackers, the packet will not contain the path traveled between the two attackers. This would falsely make the receiver conclude that the path containing the attackers is the shortest path available.

4.5 Denial of Service (DoS)

In the DoS attack [15], an attacker explicitly attempts to prevent legitimate users from using system services. This type of attack impacts the availability of the system. An ad hoc wireless network is vulnerable to DoS attacks because of its dynamic changing topology and distributed protocols. Examples of DoS attacks include:

Consumption of Scarce Resources
Attacker can consume valuable network resources (e.g. bandwidth, memory and access points) so that the entire network becomes unavailable to users.

Destruction or Alteration of Configuration Information
In this DoS attack, an attacker attempts to alter or destroy configuration information, thus preventing legitimate users from using the network. An

improperly configured network may not perform well or may not operate at all.

5 Security Mechanisms and Solutions for Routing Protocols in Ad Hoc Wireless Networks

Message encryption and digital signatures are two important mechanisms for data integrity and user authentication.

There are two types of data encryption mechanisms, symmetric and asymmetric (or public key) mechanisms. Symmetric cryptosystems use the same key (the secret key) for encryption and decryption of a message, and asymmetric cryptosystems use one key (the public key) to encrypt a message and another key (the private key) to decrypt it. Public and private keys are related in such a way that only the public key can be used to encrypt messages and only the corresponding private key can be used for decryption purpose. Even if attacker comprises a public key, it is virtually impossible to deduce the private key.

Any code attached to an electronically transmitted message that uniquely identifies the sender is known as digital code. Digital signatures are key component of most authentication schemes. To be effective, digital signatures must be non-forgeable. Hash functions are used in creation and verification of a digital signature. It is an algorithm which creates a digital representation or *fingerprint* in the form of a *hash value* (or *hash result*) of a standard length which is usually much smaller than the message and unique to it. Any change to the message will produce a different hash result even when the same hash function is used. In the case of a secure hash function, also known as a *one-way hash function*, it is computationally infeasible to derive the original message from knowledge of its hash value.

In ad hoc wireless networks, the secrecy of the key does not ensure the integrity of the message. For this purpose, Message Authentication Code (MAC) [1] is used. It is a hashed representation of a message and even if MAC is known, it is impractical to compute the message that generated it. A MAC, which is a cryptographic checksum, is computed by the message initiator as a function of the secret key and the message being transmitted and it is appended to the message. The recipient re-computes the MAC in the similar fashion upon receiving the message. If the MAC computed by the receiver matches the MAC received with the message then the recipient is assured that the message was not modified.

Next, we present security mechanisms specifically tailored for specific routing mechanisms.

5.1 Secure Efficient Ad hoc Distance Vector (SEAD)

Secure Efficient Ad hoc Distance Vector (SEAD) [7] is a proactive routing protocol, based on the design of DSDV [19]. Besides the fields common with DSDV, such as destination, metric, next hop and sequence number, SEAD routing tables maintain a hash value for each entry, as described below. This paper is concerned with protecting routing updates, both periodic and triggered, by preventing an attacker to forge better metrics or sequence numbers in such update packets.

The key feature of the proposed security protocol is the use one-way hash chains, using an one way hash function H. Each node computes a list of hash values $h_0, h_1, \cdots, h_n$, where $h_i = H(h_{i-1})$ and $0 < i \leq n$, based on an initial random value h_0. The paper assumes the existence of a mechanism for distributing h_n to all intended receivers. If a node knows H and a trusted value h_n, then it can authenticate any other value h_i, $0 < i \leq n$ by successively applying the hash function H and then comparing the result with h_n.

To authenticate a route update, a node adds a hash value to each routing table entry. For a metric j and a sequence number i, the hash value h_{n-mi+j} is used to authenticate the routing update entry for that sequence number, where $m-1$ is the maximum network diameter. Since an attacker cannot compute a hash value with a smaller index than the advertised value, he is not able to advertise a route to the same destination with a greater sequence number, or with a better metric.

SEAD provides a robust protocol against attackers trying to create incorrect routing state in other node by modifying the sequence number or the routing metric. SEAD does not provide a way to prevent an attacker from tampering next hop or destination field in a routing update. Also, it cannot prevent an attacker to use the same metric and sequence number learned from some recent update message, for sending a new routing update to a different destination.

5.2 ARIADNE

ARIADNE [8], an efficient on-demand secure routing protocol, provides security against arbitrary active attackers and relies only on efficient sym-

metric cryptography. It prevents attackers from tampering uncompromised routes consisting of uncompromised nodes.

ARIADNE ensures point-to-point authentication of a routing message by combining a shared key between the two parties and MAC. However, for secure authentication of a routing message, it relies on the TESLA [21] (see section 5.11) broadcast authentication protocol.

Design of ARIADNE is based on DSR (see section 2.1). Similar with DSR, it consists of two basic operations, route discovery and route maintenance. ARIADNE makes use of efficient combination of one way hash function and shared keys. It assumes that sender and receiver share secret (non-TESLA) keys for message authentication. The initiator (or sender) includes a MAC computed with an end-to-end key and the target (or destination) verifies the authenticity and freshness of the request using the shared key. Pre-hop hashing mechanism, a one-way hash function that verifies that no hop is omitted, is also used in Ariadne. In the case of any dead link, a *RouteError* message is sent back to the initiator. Errors are generated just as regular data packets and intermediate nodes remove routes that use dead links in the selected path.

ARIADNE provides a strong defense against attacks that modify and fabricate routing information. When it is used with an advanced version of TESLA called TIK (see section 5.9), it is immune to wormhole attacks. However, it is still vulnerable to selfish node attack. General security mechanisms are very reliable but key exchanges are complicated, making ARIADNE infeasible in the current ad hoc environments.

5.3 Security Aware Routing (SAR)

Security Aware Routing (SAR) [13] is an on demand routing protocol based on AODV (see section 2.2). It integrates the trust level of a node and the security attributes of a route to provide an *integrated security metric* for the requested route. By incorporating a Quality of Protection (QoP) as a routing metric, the route discovery can return quantifiable secure routes. The QoP vector used is a combination of security level and available cryptographic techniques

SAR introduces the notion of a *trust hierarchy*, where nodes of the ad hoc wireless network are divided into different trust levels such that an initiator can impose a minimum trust level for all the nodes participating in the source-destination communication. Note that a path with the required trust level might not exist even if the network is connected. Even if SAR

discovers fewer routes than AODV, they are always secured.

The initiator of the route in SAR includes a security metric in the route request. This security metric is the minimum trust level of the nodes that can participate in the route discovery. Consequently, only those nodes that have this minimum security level can participate in the route discovery. All other nodes that are below that trust level will drop the request packets. If an end-to-end path with the required security is found, the intermediate node or destination sends a suitably modified *RouteReply*. In the case of multiple paths satisfying the required security attributes, SAR selects the shortest such route. If route discovery fails, then a message can be sent to the initiator so that it can lower the trust level.

In the case of a successful path search, SAR always finds a route with quantifiable guarantee of security. This can be done by having nodes of a trust level share a key. Thus, a node that does not have a particular trust level will not possess the key for that level, and as a result it will not be able to decrypt the packets using the key of that level. Therefore, it will not have any other option but to drop the packet.

SAR uses sequence numbers and timestamps to stop replay attacks. Threats like interception and subversion can be prevented by trust level key authentication. Modification and fabrication attacks can be stopped by verifying the digital signatures of the transmitted packets.

One of the main drawbacks of using SAR is the excessive encrypting and decrypting required at each hop during the path discovery. In a mobile environment, the extra processing leads to an increased power consumption.

A route discovered by SAR may not be the shortest route in terms of hop-count, but it is secure. Such a path ensures that only the nodes having the required trust level will read and re-route the packets, but at the same time malicious node can steal the required key, a case in which the protocol is still open for all kinds of attacks.

5.4 Secure Routing Protocol (SRP)

Secure Routing Protocol (SRP) [17], is another protocol extension that can be applied to many of the on demand routing protocols used today. SRP defends against attacks that disrupt the route discovery process and guarantees to identify the correct topological information.

The basic idea of SRP is to set up a security association (SA) between a source and a destination node without the need of cryptographic validation of the communication data by the intermediate nodes. SRP assumes that

this SA can be achieved through a shared key K_{ST} between the source S and target T. Such a security association should exist priori to the route initiation phase.

The source S initiates the route discovery by sending a route request packet to the destination T. The SRP uses an additional header called SRP header to the underlying routing protocol (e.g. AODV) packet. SRP header contains the following fields: the query sequence number Q_{SEC}, query identifier number Q_{ID}, and a 96 bit MAC field.

Intermediate nodes discard a route request message if SRP header is missing. Otherwise, they forward the request towards destination after extracting Q_{ID}, source, and destination address. Highest priority is given to nodes that generate requests at the lowest rates and vice versa.

When the target T receives this request packet, it verifies if the packet has originated from the node with which it has SA. If Q_{SEC} is greater or equal to Q_{MAX}, the request is dropped as it is considered to be replayed. Otherwise it calculates the keyed hash of the request fields and if the output matches SRP MAC then authenticity of the sender and integrity of the request are verified.

On the reception of a route reply, S checks the source address, destination addresses, Q_{ID}, and Q_{SEC}. It discards the route reply if it does not match the currently pending query. In case of a match, it compares reply IP source-route with the exact reverse of the route carried in reply packet. If the two routes match then S calculates the MAC by using the replied route, the SRP header fields, and the secure key between source and destination. If the two MAC match then the validation is successful and it confirms that the reply did came from the destination T.

SRP suffers from the lack of validation mechanism for route maintenance messages as it does not stop a malicious node from harming routes to which that node already belongs to. SRP is immune to IP spoofing because it secures the binding of the MAC and IP address of the nodes but it is prone to wormhole attacks and invisible node attacks.

5.5 Secure Routing Protocol for Ad Hoc Networks (ARAN)

A Secure Routing Protocol for Ad Hoc Networks (ARAN) [22] is an on-demand protocol designed to provide secure communications in managed-open environments. Nodes in a managed-open environment exchange initialization parameters before the start of communication. Session keys are exchanged or distributed through a trusted third party like a certification

authority.

Each node in ARAN receives a certificate after securely authenticating its identity to a trusted certificate server T. Nodes use these certificates to authenticate themselves to other nodes during the exchange of routing messages. The certificate contains the node's IP address, its public key, as well as the time of issuing and expiration. These fields are concatenated and signed by the server T. A node A receives a certificate as: $T \rightarrow A : cert_A$ $=[IP_A, K_{A+}, \text{t}, \text{e}]\ K_{T-}$.

In the authentication phase, ARAN ensures the existence of a secure path to the destination. Each intermediate node in the network stores the route pair (previous node, the destination node). All the fields are concatenated and signed with source node I's private key. A combination of the nonce number (N_I) and timestamp (t) is used to obtain data freshness and timeliness property. Each time I performs a route discovery, it monotonically increases the nonce. The signature prevents spoofing attacks that may alter the route or form loops. Source node I broadcasts a Route Discovery Packet (RDP) for a destination D as $I \rightarrow brdcst :[RDP, IP_D, cert_I, N_I, \text{t}]K_{I-}$.

Each node that receives the RDP for the first time removes any other intermediate node's signature, signs the RDP using its own key, and broadcasts it to all its neighboring nodes. This continues until destination node D eventually receives the packet.

After receiving the RDP, the destination node D sends a Reply (REP) packet back along the reverse path to the source node I. If J is the first node on the reverse path, REP packet is sent as $D \rightarrow J :[\ REP, IP_I, cert_D, N_I, \text{t}]\ K_{D-}$.

When the source node I receives the REP packet, it verifies the destination's signature K_{D-} and nonce N_I. When there is no traffic on an existing route for some specific time, then that route is deactivated in the routing table. Nodes use an ERR message to report links in active routes broken due to node movement.

Using pre-determined cryptographic certificates, ARAN provides network services like authentication and non-repudiation. Simulations show that ARAN is efficient in discovering and maintaining routes but routing packets are larger in size and overall routing load is high. Due to heavy asymmetric cryptographic computation, ARAN has higher cost for route discovery. It is not immune to wormhole attack and if nodes do not have time synchronization, then it is prone to replay attacks as well.

5.6 Security Protocols for Sensor Network (SPINS)

Security Protocols for Sensor Network (SPINS) [25] is a suite of two security building blocks which are optimized for ad hoc wireless networks. It provides important network services like data confidentiality, two party data authentication, and data freshness through Secure Network Encryption Protocol (SNEP) and secure broadcast through Micro Timed Efficient Stream Loss-tolerant Authentication (μTESLA).

Most of the current protocols are not practical for secure broadcast as they use asymmetric digital signatures. These signatures have high cost of creation and verification. SPINS introduces μTESLA (see section 5.11), an enhanced version of TESLA which uses symmetric cryptographic techniques for authentications and asymmetry cryptography only for the delayed disclosure of keys. Tight lower bound on the key disclosure delay and robustness against DoS attacks makes μTESLA a very efficient and secure protocol for data broadcast.

SNEP provides point to point communication in the wireless network. It relies on a shared counter between a sender and a receiver in order to ensure semantic security. Thus it protects message contents of encrypted messages from eavesdroppers. Since both nodes share the counter and increment it after each block, the counter does not need to be sent with the message. In this way, the same message is encrypted differently each time. A receiver node is assured that the message originated from the legitimate node if the MAC verifies successfully. The counter value in the MAC eliminates replaying of old messages in the network.

SPINS is the first secure and lightweight broadcast authentication protocol. The computation costs of symmetric cryptography are low and the communication overhead of 8 bytes per message is almost negligible when compared to the size of a message. SNEP ensures semantic security, data authentication, replay protection, and message freshness whereas μTESLA provides authentication for secure data broadcast.

5.7 Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks (CONFIDANT)

Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks (CONFIDANT) [2] protocol is designed as an extension to reactive source-routing protocol such as DSR. It is a collection of components which interact with each other for monitoring, reporting, and establishing routes by avoiding

misbehaving nodes. CONFIDANT components in each node include a network monitor, reputation system, trust manager, and a path manager.

Each node in this protocol monitors their neighbors and updates the reputation accordingly. If they detect any misbehaving or malicious node, they can inform other *friend* nodes by sending an ALARM message. When a node receives such an ALARM either directly from another node or by listening to the ad hoc network, it calculates how trustworthy the ALARM is based on the source of the ALARM and the total number of ALARM messages about the misbehaving node.

Trust manager sends alarm messages to other nodes to warn them of malicious nodes. Incoming alarms are checked for trustworthiness. Trust manager contains an alarm table, trust level table and a *friend* list of all trust worthy nodes to which a node will send alarms.

Local rating lists and black lists are maintained in the reputation system. These lists are exchanged with friend nodes and timeouts are used to avoid old lists. A node gives more importance to its own experience than to those events which are observed and reported by others. Whenever the threshold for certain behavior is crossed, path manager does the re-ranking by deleting the paths containing malicious nodes and ignoring any request from misbehaving nodes. At the same time, it sends an alert to the source of the path so that it can discover some other route.

When DSR is fortified with the CONFIDANT protocol extensions, it is very scalable in terms of the total number of nodes in the network and it performs well even if more than 60% of the nodes are misbehaving. The overhead for incorporating different security components is manageable for ad hoc environment. However, detection based reputation system has few limitations and routes are still vulnerable to spoofing and Sybil attacks.

5.8 Defense Mechanisms Against Rushing Attacks

Rushing attacks [9] (see section 4.3) are mostly directed against on demand routing protocols such as DSR. To counter such attacks, a generic secure route discovery component called Rushing Attack Prevention (RAP) is used. RAP combines the following mechanisms: *Secure Neighbor Detection*, *Secure Route Delegation*, and *Randomized Route Request Forwarding*. Any on demand routing protocol such as ARIADNE can be used as underlying protocol to RAP.

In *Secure Neighbor Detection*, a three round mutual authentication procedure is used between a sender and a receiver to check if they are within

normal communication range of each other. First, a node forwards a *Neighbor Solicitation* packet to the neighboring node which replies with a *Neighbor Reply* packet and finally, the initial node sends *Neighbor Verification* packet to confirm that both nodes are neighbors.

Secure Route Delegation verifies that all the steps in *Secure Neighbor Detection* phase were carried out. Before sending a route update to its neighbor, it signs a route attestation, delegating the rights to the neighbor to further propagate the update.

In *Randomize Message Forwarding*, a node buffers k route requests and then it randomly forwards only one of these k requests. By limiting the total number of requests sent by a node, it prevents flood attacks in the network. Each request carries the list of all the nodes traversed by that request. Furthermore, bi-directional verification is also used to authenticate the neighbors.

By using efficiently combining these three mechanisms, RAP can find usable routes when other protocols cannot. When it is enabled, it has higher overhead than other protocols, but currently it is the only protocol that can defend against rushing attacks. However, network is still prone to rushing attacks if an attacker can compromise k nodes.

5.9 Defense Mechanisms Against Wormhole Attacks

In order to prevent the wormwhole attacks (see section 4.4), the *packet leashes* mechanism [11] proposes to add additional information (referred as *leashes*) to the packets in order to restrict packet's maximum allowed transmission distance.

Geographical leash and *temporal leash* can be used to detect and stop wormhole attacks. Geographical leash insures that the recipient of the packet is within a certain distance from the sender while temporal leash is used to enforce an upper bound on the packet's life time, thus restricting packet's maximum travel distance. Temporal leash uses packet's expiration time to detect a wormhole. The expiration time is computed based on the allowed maximum transmission distance and the speed of light. A node will not accept any packet if this expiration time has passed.

TIK (TESLA with Instant Key Disclosure) protocol is an extension of TESLA (see section 5.11) and it is implemented with temporal leashes to detect wormholes. It requires each communicating node to know one public key for each other node in the network. The TIK protocol uses an efficient mechanism *Merkle Hash* tree [10] for key authentication. The root value m

of the resulting hash tree commits to all the keys and is used to authenticate any leaf key efficiently. Hash trees are generally large so only the upper layers are stored while lower layers can be computed on demand.

The TIK packet is transmitted by sender S as $S \rightarrow R : HMAC_{K_i}(M)$, M, T, K_i, where M is the message payload, T are the tree authentication values, and K_i is the key used to generate the HMAC. After the receiver R receives the HMAC value, it uses the hash tree root m and the hash tree values T to verify that the key K_i at the end of the packet is authentic, and then uses the key K_i to verify the HMAC value in the packet. The receiver R only accepts the packet as authentic if all these verifications are successful.

A receiver can verify the TESLA security condition as it receives the packet, thereby eliminating the authentication delay of TESLA. Packet leashes are effective mechanisms, but TIK is not feasible in resource constraint networks due to the expensive cryptographic mechanisms implemented. The lack of accurate time synchronization in today's systems prevent TIK from providing a usable wormhole detection mechanism. Another potential problem with leashes using a timestamp in a packet is that the sender may not know the precise time at which it will transmit the packet and generating a digital signature in that time may not be possible.

5.10 Defense Mechanisms Against Sybil Attacks

In a Sybil attack [16] (see section 4.1), a malicious node acts on behalf of a larger number of nodes either by impersonating other nodes or simply by claiming false identities. Most of the secure protocols are prone to this type of attack. However, there are various key distribution mechanisms which can be used efficiently to defend against Sybil attacks.

Sybil nodes can carry out a variety of attacks. For example, network nodes use *voting* for many purposes. With enough Sybil nodes, an attacker may be able to determine the outcome of every vote. Sybil nodes, due to their larger number, are allocated more resources and they can create DoS for legitimate nodes. Ad hoc wireless networks can use *misbehavior detection* property to detect any malfunctioning node. An attacker with many Sybil nodes can spread the blame and pass unnoticed, having only small misbehavior actions associated with each identity.

There are a number of ways to detect Sybil attacks. In *radio resource testing*, it is assumed that nodes have only one radio and are not capable of sending or receiving on more than one channel. If a node wants to verify

whether its neighbors are Sybil nodes, then it assigns to each of its neighbors a different channel to broadcast messages. Then the node listens to one of the channels. If a message is received, this is an indication of a legitimate neighbor, whereas an idle transmission is an indication of a Sybil node.

A more authentic way of defending against Sybil attacks is *random key predistribution*. A random set of keys are assigned to each node and then every node can compute the common keys it shares with its neighbors. If two nodes share q common keys, they can establish a secure link. An *one-way Pseudo Random hash Function* (PRF) is used for validation. Thus, an attacker can not just gather a bunch of keys and claim an identity since PRF is an one way hash function.

There are two types of key distribution mechanisms [5] to counter Sybil attacks. In *single-space pairwise key distribution*, each pair of nodes is assigned a unique key. A node i stores unique public information U_i and private information V_i. The node i computes its key from $f(V_i, U_j)$ where U_j is the public key of neighboring node j. Validation is successful if a node has the pairwise key between itself and the verifier. In *multi-space pairwise key distribution*, each node is assigned, by the network, k out of m random key spaces. If two neighboring nodes have at least one key space in common, then they can compute their pairwise secret key using the corresponding single space scheme.

This is the first work that proposes various defense mechanisms against the Sybil attacks, such as *radio resource testing* and *random key predistribution*. *Random key predistribution* is already required in many applications to secure radio communication. The most effective against Sybil attacks is the *multi-space pairwise key distribution* mechanism.

5.11 Security Mechanisms for Broadcast Operation

Timed Efficient Stream Loss-tolerant Authentication (TESLA) [21] is an efficient broadcast authentication protocol with low communication and computation overhead. It can scale to large numbers of receivers, can tolerate packet loss, and uses loose time synchronization between sender and receivers.

TESLA mainly uses purely symmetric cryptographic functions, however, it achieves asymmetric properties from clock synchronization and delayed key disclosure. In this way, it does not require to compute expensive one-way functions. For this purpose, it needs sender and receivers to be loosely time-synchronized and for a secure authentication, either the receiver or the

sender must buffer some messages.

For secure broadcasting, a sender chooses a random initial key K_N and generates a one-way key chain by repeatedly computing the one-way hash function H on the starting value $K_{N-1} = H[K_N]$, $K_{N-2} = H[K_{N-1}]$, ..., $K_0 = H[K_1]$. In general, $K_i = H[K_{i+1}] = H^{N-i}[K_N]$ where $H^i[x]$ is the result of applying the function H to x, for i times.

The sender node predetermines a schedule at which it discloses each key of its one-way key chain. Keys are disclosed in the reverse order from generation, i.e. $K_0, K_1, K_2, \ldots, K_N$ then the MAC computed using the key K_i is added to the packet. When the packet reaches the receiver, it checks the security condition of the key disclosure. If the key K_i used to authenticate the packet was not disclosed, then it buffers the packet and waits for the sender to disclose K_i, while using an already disclosed key to authenticate the buffered packets. However, if the key is already disclosed, then receiver will discard the packet.

Even though TESLA is efficient, it still has few drawbacks. It authenticates the initial packet with a digital signature which is too expensive for wireless nodes and disclosing a key in each packet requires too much energy for sending and receiving. TESLA is vulnerable to DoS attacks as malicious nodes can create *buffer overflow* state in the receiver while it waits for the sender to disclose its keys.

SPINS [25] introduces Micro Timed Efficient Stream Loss-tolerant Authentication (μTESLA), a modified version of TESLA which only uses symmetric mechanisms for packet authentication and it discloses the key once per epoch. μTESLA is different from TESLA as it allows a receiver to authenticate the packets as soon as they arrive and it replaces receiver buffering with sender buffering. Immediate authentication as well as buffering only at the sender makes it a secure protocol against DoS. It has very low security overhead. The computation, memory, and communication costs are also small. Since the data authentication, freshness, and confidentiality properties require transmitting only 8 bytes per message, μTESLA is considered a very effective and robust protocol for secure data broadcasting.

TESLA with Instant Key Disclosure (TIK)[11] is another protocol for secure broadcasting implemented with temporal leashes in order to detect wormholes (see section 5.9). TIK requires accurate time synchronization between all communicating parties. It works almost in the same manner as the base protocol TESLA, but in TIK the receiver can verify TESLA security condition as it receives the packet. By eliminating the authentication delay of TESLA, it allows sender to disclose the key in the same packet. TIK is

therefore a more robust protocol than TESLA since it eliminates the waiting time imposed by disclosing the keys only after the packet was received.

6 Conclusions

Achieving a secure routing protocol is an important task that is being challenged by the unique characteristics of an ad hoc wireless network. Traditional routing protocols fail to provide security, and rely on an implicit trust between communicating nodes.

In this chapter we discuss security services and challenges in an ad hoc wireless network environment. We examine and classify major routing attacks and present a comprehensive survey on the state-of-the-art mechanisms and solutions designed to defeat such attacks. A summary of the secure routing mechanisms surveyed is presented in Table 1. The current security mechanisms, each defeats one or few routing attacks. Designing routing protocols resistant to multiple attacks remains a challenging task.

Protocol	*Security Mechanisms*	*Attacks Prevented*	*Comments*
SEAD [7]	- One-way hash chains	- Prevents an attacker from forging better metrics or sequence numbers in routing update packets	- Used with DSDV - Designed to protect routing update packets - Does not prevent an attacker from tampering other fields or from using the learned metric and sequence number for sending new routing updates
Ariadne [8]	- One-way hash chains	- Prevents attackers from tampering uncompromised routes consisting of uncompromised nodes - Immune to wormhole attack	- Used with DSR - Provides a strong defense against attacks that modify and fabricate routing information - Prone to selfish node attack
SAR [13]	- Quality of Protection (QoP) metric	- Uses sequence numbers and timestamps to stop replay attacks in routing update packets	- Used with AODV - Route discovered may not be the shortest route in terms of hop-count, but it is always secured - Defends gainst modification and fabrication attacks
SRP [17]	- Secure certificate server	- Defends against attacks that disrupt the route discovery process and guarantees to identify the correct topological information	- Used with DSR, ZRP - Lack of validation mechanism for route maintenance messages - Prone to wormhole attacks and invisible node attacks

ARAN [22]	- Secure certificate server	- Provides network services like authentication and non-repudiation	- Used with AODV, DSR - Heavy asymmetric cryptographic computation - Prone to wormhole attack if accurate time synchronozation is not available
CONFIDANT [2]	- Monitor - Reputation System - Path Manager - Trust Manager	- Attacks on packet forwarding and routing are defended efficiently	- Used with DSR - Detection based reputation system has few limitations - Vulnerable to spoofing and sybil attacks
Rushing Attacks and Defenses [9]	- Secure Neighbor Detection - Secure Route Delegation - Randomized Route Request Forwarding	- By limiting the total number of requests sent by a node and random forwarding, it prevents rushing attack to a certain level	- Used with DSR, ARIADNE - Network is still prone to rushing attacks if an attacker can compromise k nodes - Higher overhead than other protocols, but currently it is the only protocol that can defend against rushing attacks
Wormhole Attacks and Defenses [11]	- Packet Leashes - Merkle Hash Tree - One-way Hash Chains	- TIK when implemeted with packet leaches, effectively stops wormhole and DoS attacks	- Not feasible in resource constraint networks due to the expensive cryptographic mechanisms implemented - Accurate time synchronization is not easy to obtain
Sybil Attacks and Defenses [16]	- Radio Resource Testing - Random Key Predistribution - one-way Pseudo Random Hash Function	- Multi-space Pairwise Key Distribution is most effective mechanism against sybil attack	- First work that proposes various defense mechanisms against the Sybil attacks
TESLA [21]	- One-way Hash Chain	- Uses loose time synchronization and delayed time synchronization to provide secure broadcast	- Vulnerable to DoS attacks as malicious nodes can create buffer overflow state - Accurate time synchronization is not easy to obtain

Table 1: Comparison and summary of different routing security mechanisms.

References

[1] J. Barkley, NIST Special Publication: Symmetric Key Cryptography, http://csrc.nist.gov/publications/nistpubs/800-7/node208. html

[2] S. Buchegger and J. L. Boudec, Performance Analysis of the CONFIDANT Protocol Cooperation Of Nodes Fairness In Dynamic Ad-hoc NeTworks, *In Proc. of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, Jun. 2002.

[3] T. H. Clausen, G. Hansen, L. Christensen, and G. Behrmann, The Optimized Link State Routing Protocol, Evaluation Through Experiments and Simulation, *Proc. of IEEE Symp. on Wireless Personal Mobile Communications 2001*, Sep. 2001.

[4] D. Coppersmith and M. Jakobsson, Almost Optimal Hash Sequence Traversal, *In Proc. of The Sixth Intl. Conf. on Financial Cryptography (FC 2002), Lecture Notes in Computer Science*, Springer 2002.

[5] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, A Pairwise Key Predistribution Scheme for Wireless Sensor Networks, *ACM CCS 2003*, Oct. 2003, pp. 42-51.

[6] Z. J. Haas, The Routing Algorithm for the Reconfigurable Wireless Networks, *Proc. of ICUPC 1997*, Vol 2, Oct. 1997, pp. 562-566.

[7] Y. -C. Hu, D. B. Johnson and A. Perrig, SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks, *Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'02)*, Jun. 2002.

[8] Y. -C. Hu, D. B. Johnson, and A. Perrig, Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks, *Mobicom'02*, 2002.

[9] Y. -C. Hu, D. B. Johnson, and A. Perrig, Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols, *WiSe 2003*, 2003.

[10] Y. -C. Hu, D. B. Johnson, and A. Perrig, Efficient Security Mechanisms for Routing Protocols, *The 10th Annual Network and Distributed System Security Symp. (NDSS)*, Feb. 2003.

[11] Y. -C. Hu, A. Perrig, and D. B. Johnson, Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks, *Infocom* 2003.

[12] D. B. Johnson and D. A. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, *Mobile Computing, Kluwer Academic Publishers*, Vol 353, 1996, pp. 153-181.

[13] R. Kravets, S. Yi, and P. Naldurg, A Security-Aware Routing Protocol for Wireless Ad Hoc Networks, *In ACM Symp. on Mobile Ad Hoc Networking and Computing*, 2001.

[14] S. Murthy and J. J. Garcia-Luna-Aceves, An Efficient Routing Protocol for Wireless Networks, *ACM Mobile Networks and Applications Journal, Special Issue on Routing in Mobile Communication Networks*, Vol 1, No 2, Oct. 1996, pp. 183-197.

[15] C. S. R. Murthy and B. S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Prentice Hall PTR, 2004.

[16] J. Newsome, E. Shi, D. Song, and A. Perrig, The Sybil Attack in Sensor Networks: Analysis & Defenses, *Proc. of the 3rd Intl. Symp. on Information Processing in Sensor Networks*, 2004.

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

[18] V. D. Park and M. S. Corson, A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, *IEEE Infocom 1997*, Apr. 1997, pp. 1405-1413.

[19] C. E. Perkins and P. Bhagwat, Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, *SIGCOMM'94 Conf. on Communications Architectures, Protocols and Applications*, Aug. 1994, pp. 234-244.

[20] C. E. Perkins and E. M. Royer, Ad Hoc On-Demand Distance Vector Routing, *IEEE Workshop on Mobile Computing Systems and Applications 1999*, Feb. 1999, pp. 90-100.

[21] A. Perrig, R. Canetti, D. Tygar, and D. Song, The TESLA Broadcast Authentication Protocol, RSA Cryptobytes *(RSA Laboratories)*, Vol 5, No 2, Summer/Fall 2002, pp. 2-13.

[22] 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 Intl. Conf. on Network Protocol (ICNP)*, Nov. 2002.

[23] P. Sinha, R. Sivakumar, and V. Bharghavan, CEDAR: A Core Extraction Distributed Ad Hoc Routing Algorithm, *IEEE Journal On Selected Areas in Communications*, Vol 17, No 8, Aug. 1999, pp. 1454-1466.

[24] W. Stallings, Cryptography and Network Security: Principles and Practices, 3rd edition, *Prentice Hall*, 2003.

[25] R. Szewczyk, V. Wen, D. Culler, J. D. Tygar, and A. Perrig, SPINS: Security Protocols for Sensor Networks, *In Seventh Annual ACM Intl. Conf. on Mobile Computing and Networks (Mobicom 2001)*, 2001.

[26] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, The Broadcast Storm Problem in a Mobile Ad Hoc Network, *ACM Wireless Networks*, Vol 8, No 2, Mar. 2002, pp. 153-167.

[27] J. Wu and F. Dai, A Generic Distributed Broadcast Scheme in Ad Hoc Wireless Networks, *IEEE Transactions on Computers*, Vol 53, No 10, Oct. 2004, pp. 1343-1354.