

Security Threats and Solutions in MANETs: A Case Study using AODV and SAODV

Jan von Mulert, Ian Welch*, Winston K.G. Seah**

*School of Engineering and Computer Science, Victoria University of Wellington
P.O. Box 600, Wellington 6140, New Zealand*

Abstract

Mobile ad hoc network (MANET) security has become the focus of prolific research efforts. Driven by the unique and considerable difficulties of providing security arising from the dynamic nature of MANETs, many security schemes have been proposed. Rather than trying to encompass the entire field of MANET security, this paper focuses on networks using the popular Ad-hoc On-demand Distance Vector (AODV) protocol and a secure extension to AODV, the Secure AODV (SAOV) protocol. SAODV is representative of a number of secure versions of the AODV protocol in that it relies upon the use of cryptographic mechanisms protect the routing control messages of AODV from being forged and/or altered by attackers. We conduct a vulnerability analysis of SAODV to identify unresolved threats to the algorithm, such as, medium access control layer misbehaviour, resources depletion, blacks holes, wormholes, jellyfish and rushing attacks. We then compare this vulnerability analysis to schemes that have been proposed to combat the identified threats. These proposals include multipath routing, incentive schemes, directional antennae, packet leashes, randomized route requests, localized self-healing communities and a reactive intrusion detection node blacklisting scheme.

Keywords: mobile ad hoc networks, security, AODV, SAODV

1. Introduction

Mobile Ad-hoc Networks (MANETs) have unique characteristics that make securing data flows within the network a non-trivial challenge [1]. These characteristics include: open peer-to-peer architecture, shared wireless medium, stringent resource constraints, highly dynamic network topology and nodes' openness to physical capture. Among the many routing protocols that have been developed to allow self-configuring and self-maintaining routing in MANETs, the Ad-hoc On-demand Distance Vector (AODV) protocol has emerged to be one of the most popular. It has been the subject of much academic research and has been ratified as an experimental RFC [2] by the Internet Engineering Task Force (IETF). For these reasons, we choose AODV as the example protocol for security analysis. This protocol was developed under the assumption that all nodes in the network are friendly and cooperative. Consequently, there are many attack vectors in AODV which allow attackers to disrupt its route discovery and packet forwarding processes. Various extensions to secure AODV have been proposed, like Secure AODV (SAODV) [3], ARAN [4], SEAR [5] and SEAODV [6].

While AODV security has been well researched, we aim to review whether this problem has been adequately addressed. SAODV is a good representation of the cryptographic approach to securing AODV, using digital signatures to protect the integrity of routing data. We show that SAODV is still vulnerable (and by extension, other protocols also relying upon cryptographic protection of routing data), especially to more sophisticated insider attacks, rushing/tunneling attacks and medium access control (MAC) layer misbehaviour. We evaluate proposals that combat these vulnerabilities, including modifications to SAODV, detection/blacklisting schemes, aiding routing with redundant nodes, flow based access control, and multipath routing. However, these proposals each only provide solutions to one or two vulnerabilities in a 'piece meal' fashion. We aim to determine if any vulnerabilities remain unaddressed, and identify areas needing future research. We then summarize our key findings before concluding the paper.

1.1. AODV Overview

AODV is a reactive or on-demand routing protocol which means a route between two nodes will be determined only when there is data to be transmitted. Each node's routing table only contains the next hop to a particular destination, so the information on the route to be traversed by a packet is distributed along all the nodes on the path. Neighbour connectivity is established with periodic

*Corresponding author

**Principal corresponding author

*Email addresses: vonmulert@gmail.com (Jan von Mulert),
Ian.Welch@vuw.ac.nz (Ian Welch), Winston.Seah@ecs.vuw.ac.nz
(Winston K.G. Seah)*

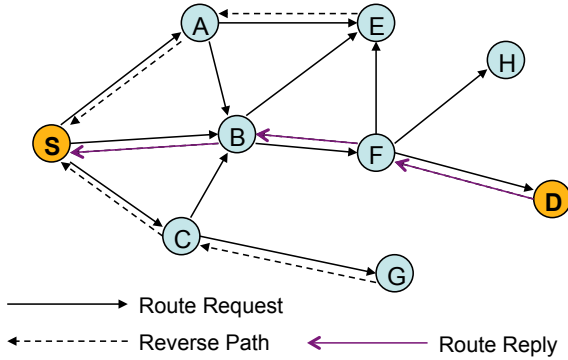


Figure 1: Route Discovery in AODV, Route Requests are Flooded through the network, the first to reach the destination node generate a route reply which is unicast back to the source.

Hello Messages. Routes are found by flooding of Route Request (RREQ) messages Figure 1. As node each node receives and retransmits the RREQ it records the previous hop in its routing table. A unicast Route Reply (RREP) is sent back from the destination or any node with a route to the destination. Nodes route this RREP back to the source using the previous hop information recorded from the RREQ. On the way the RREPs previous hop information is recorded into each nodes routing table. setting up the final path to the destination. RREQs use hop counts and RREP use destination sequence numbers (DSNs) so the sender can differentiate routes based on hop count and freshness. Route maintenance is done using Route Error (RERR) messages. On sensing a broken link in an active route (failing to receive regular HELLO messages from a node), a RERR message is sent to its upstream neighbours that use it as the next hop in the broken route(s).

1.2. AODV Vulnerabilities

AODV has no security mechanism and is vulnerable to many kinds of attacks that manipulate its routing control mechanisms. A comprehensive analysis of possible misuses of the AODV protocol has been presented in [7]. They classify misuse into atomic misuse (indivisible manipulation of one routing message) and compound misuse (combination of atomic misuses or legitimate messages to achieve misuse goals). We have created a synthesis of misuse goals and type of misuse to help us to identify commonalities between attacks. This is shown in Table 1 summarises possible attacks on AODV and is based on the synthesis of an analysis of potential attacker misuse goals [7] and attacks [8]. The columns are based on the types of attack outlined in [8] while the rows specify the misuse goals outlined in [7], with the cells explaining how the type of attack is conducted using atomic and composite misuse to achieve the misuse goals.

2. Securing AODV

There have been various proposals to improve the security of AODV and these include SAODV [3], ARAN [4],

SEAR [5] and SEAODV [6]. As an example of the limitations of secure routing protocols, we will look at SAODV. SAODV uses asymmetric cryptography to secure AODV routing messages against manipulation. Providing authenticity, integrity and confidentiality to data packets is outside the scope of SAODV, while secure end-to-end communication is usually achieved using cryptographic secure protocols such as IPsec. In RFC3561 [2], the IPsec Authentication Header (AH) has been proposed as a way to protect routing control messages where appropriate security associations exist between nodes. However, this has been dismissed as a “canned” solution, that may not be directly applicable to MANETs [8].

The setting up of these security associations in SAODV means distribution of public and private keys and implicit to this is establishing the identity of the node the keys are given to. Everything is considerably simplified if the participants within the network are identified beforehand so they can be manually pre-configured with cryptographic keys. Where network membership is dynamic, key distribution to new nodes and establishing node identity is complicated. The use of a trusted third party violates the decentralized nature of MANETs and much work has gone into developing proper de-centralized key distribution methods.

In SAODV, digital signatures are used to authenticate non-mutable fields in AODV’s routing control messages, and this prevents impersonation of source nodes (sending RREQ) and destination nodes (sending RREP) [9]. To prevent an illegitimate RREP being sent by an intermediate node, the protocol is either altered such that RREPs can only be sent by destination nodes or a double signature extension is added to RREPs (containing the intermediate node’s and destination’s signatures) [9]. The use of cyclic sequence numbers is removed, once the maximum sequence number is reached a node should get a new key pair to prevent replay attacks (if key distribution is possible). The mutable hop count is protected by hash chains, as each node increments the hop count its digital signature is added to the hash chain, preventing nodes from decrementing the hop count. Hop count protection is not perfect because it does not prevent a malicious node from leaving the hop count unchanged [9]. Nodes sign the entire RERR message to prevent tampering and impersonation. Furthermore, the protocol is modified such that nodes no longer update their DSN from RERRs or RREPs, preventing attacks that manipulate DSNs.

The kind of attacks that are prevented by SAODV are limited to those involving impersonation of nodes or manipulation of routing control messages, and it remains vulnerable to insider attacks from authenticated nodes which have been compromised or physically captured (not unlikely given many wireless ad-hoc nodes’ portable nature.) The insider node problem is beyond the scope of proactive security schemes like SAODV, and motivates the need more in depth security mechanisms. Zapata who proposed SAODV has stressed that it is not a suitable protocol for

Table 1: AODV Vulnerabilities. In the table below, MN is a malicious node and DSN is a destination sequence number.

Type of Misuse				
	Drop:	Modify & Forward:	Forged Reply:	Active Forge:
Misuse Goals <u>Route Disruption</u> Disrupting routing tables and breaking links	Routing or data packets are dropped without forwarding Dropping RREQ, RREP or data messages	MN incorrectly modifies routing control message and forwards it	MN sends forged routing control message in response to real message	MN sends forged route control message without first receiving a real one MN broadcasts forged RERR causing other nodes to delete routes from routing table; using large DSN will cause new RREQ to appear stale and be ignored.
<u>Route Invasion</u> MN attracts routes to itself and packets may now be intercepted or dropped.		MN attracts routes by forging hop count or DSN in RREP to make route appear shorter or fresher.	MN impersonates destination node by sending RREP with forged destination in response to RREQ.	MN impersonates source node with forged source in RREQ.
<u>Node Isolation</u> Attempting to isolate a node from communicating with the rest of the network.	Routing messages are forwarded while data messages are dropped; known as <i>blackhole</i> attack.	MN attracts routes through forging hop count or DSN in RREP to make route appear shorter or fresher, then drops data packets; composite misuse.	MN impersonates destination node by sending RREP with forged destination in response to RREQ, then drops data packets; composite misuse.	Routes to a node can be disrupted by sending forged RERR messages. In SAODV, where the RERR messages are authenticated, an MN can only isolate itself this way.
<u>Resource Depletion</u> A type of Denial of Service attack that attempts to use up network resources, e.g. battery power, storage space, etc.				Flooding the network with RREQs or RERRs to use up resources; sending junk data packets.

military applications [10]. However, we choose it as a representative secure extension to AODV for analysis as other notable schemes such as SEADOV, ARAN and SEAR also rely on cryptographic mechanisms to secure routing packets [6]. All the secure extensions to AODV discussed below have similar goals, ensuring integrity of mutable fields in routing packets (in ARAN’s case by removing them) and preventing forgeries by allowing intermediate nodes to check the packets’ authenticity. These protocols present an almost identical attack surface, what differentiates them is the computational power required to implement the cryptographic techniques.

ARAN uses hop-by-hop and end-to-end authentication in Route Discovery Packets (RDPs, functionally similar to RREQs) [4]. RDPs are signed by the initiator node, as in SAODV, so that no other node may forge the source of an RDP. A nonce and timestamp are used, preventing replay attacks and route loops. Unlike RREQs in SAODV, ARAN’s RDPs are signed and authenticated by each intermediate node. At each hop after the first, the signature of the previous node is authenticated, then removed and the intermediate node re-signs the RDP before rebroadcasting it. The destination sends a signed REP (Reply Packet) in response to the first RDP received. There are no mutable fields in an RDP and hop count is not used. This means that unlike the other protocols considered, ARAN has no ability to differentiate the shortest of multiple routes, only the quickest. The other major difference between SAODV and ARAN is the computationally expensive hop-by-hop authentication used by ARAN. This is supposed to prevent unauthenticated nodes from participating in the routes; however, an unauthenticated node can still insert itself into a route undetected by replaying an unaltered RDP and the corresponding REP.

The SEAR protocol, a secure extension of AODV, uses one-way hash functions to construct a set of hash values called *authenticators*, associated with each node [5]. The authenticators are used to ensure the authenticity of routing control messages. Mutable fields in the RREQs may still be altered by malicious nodes, but only to decrease the DSN or increase the hop count. Due to the properties of AODV, an attacker can obtain no benefit from such an alteration. In SAODV, a malicious node may pass on a RREQ without incrementing the hop count; this behavior is prevented in SEAR by encoding the node’s identity into the hash values to create an authenticator hash tree. RERR messages are a special case and a variation of the TESLA protocol [11] is used to secure these, requiring loose time synchronization of nodes.

SEAODV [6] protects RERR messages using the HEAP protocol [12], which unlike TESLA [11], requires no time synchronization. HEAP makes use of a bootstrap process based upon public key cryptography to create pairwise and groupwise symmetric keys for use with secure hash functions. Similar to SEAR, hash chains are used to protect routing packets, the individual values of which are also called *authenticators*. These are partitioned into

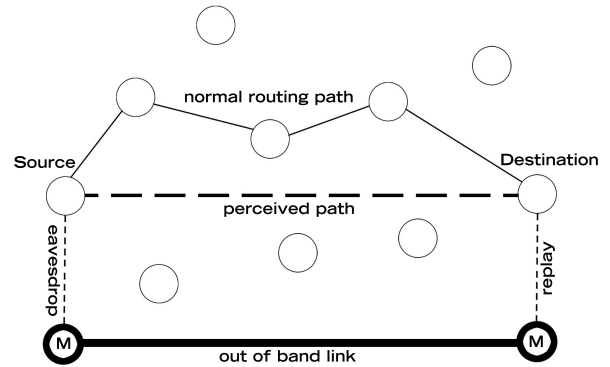


Figure 2: Wormhole Attack: Eavesdropped packets are passed between malicious nodes on an out of band link, then replayed. The source and destination will be deluded into believing they are neighbours.

separate portions to use to protect sequence numbers and hop counts. Intermediate nodes are unable to create hash values for larger sequence numbers or smaller hop counts. In small networks, the authenticity of routing control messages is safeguarded by encoding the originating node ID into the hash values, and in larger networks, the routing packets are encoded with multi-value techniques similar to those used in SEAD [13].

Not all proposals to secure AODV are based on cryptography, another technique is to use a reactive Intrusion Detection System (IDS) to identify and blacklist intruders. A soft state security proposal is presented in [14], it uses a finite state machine with a sophisticated tree structure to allow neighbours to monitor one another’s request-response flows. Another specification based intrusion detection proposal for AODV is outlined in [15]. AODVSTAT applies State Transition Analysis Techniques, first developed for wired networks, to AODV [16]. Other Ideas for MANET IDS include anomaly detection through statistical network analysis and reputation based systems where neighbours assign trust levels [14]. Comparing the effectiveness of the cryptographic approach versus the intrusion detection approach is outside the scope of this paper. It may be best to merge the two techniques and in section 4.5.1 we examine a paper [17] which designs an IDS for SAODV.

3. SAODV Vulnerabilities

This section identifies vulnerabilities that are not preventable by the straightforward application of cryptographic mechanisms. Authenticated malicious nodes can deplete network resources, carry out Blackhole and Jellyfish attacks, and block channels at the MAC layer. Wormholes, rushing attacks and some denial-of-service (DoS) attacks can even be carried out by unauthenticated nodes.

Table 2: Summary of Secure AODV Variants

Protocol Features	Secure AODV Variants			
	ARAN	SAODV	SEAR	SEAODV
Hop by Hop Authentication	•		•	◦
End to End Authentication		•		
Src Authentication with Digital Signatures	•	•		
Src Authentication with Hash Authenticators			•	•
Mutable Field Protection Hash Chains		•		
Mutable Field Protection Hash Authenticators			•	•
Hop Count Omission	•			
TESLA protection for Route Error			•	
HEAP protection for Route Error				•
Intermediate Nodes don't send Route Reply			•	

• Implements; ◦ Optional

3.1. Wormhole Attacks

Two attackers working together can create a route attracting “wormhole”, also known as a tunneling attack, which cannot be prevented by SAODV. One attacker forwards control messages only to the second attacking node, and if these messages are successfully “rushed” (through a high quality out-of-band link), it leads to a situation where nodes can only find a route to nodes on the far side of this tunnel by routing through the tunnel (see Figure 2). One node encapsulates a RREQ received into a data packet, then tunnels this to the second node (that may be several hops away) which unwraps and forwards it. This means from the point of view of the hop count, the nodes are next to each other, and the wormhole is invisible to the routing protocol. Therefore, the shortest route between nodes in the vicinity of one attacking node to nodes in the vicinity of the other attacking node will always be through the tunnel. In single path AODV, the first RREQ received by the destination is replied to with the RREP, this means the discovered route will be the quickest but not necessarily the shortest route. A wormhole therefore must deliver RREQs quicker than the legitimate paths to be effective. Once an attacker has attracted routes to itself, it can sniff data packets or drop data packets leading to DoS. Even a single unauthenticated node can insert itself into routes simply by replaying all recorded neighbour detection, routing, and data packets without altering any of them. Of course, this can be easily detected since nodes will hear their own “hello” packets being replayed.

3.2. Rushing Attacks

SAODV, ARAN [4] and also the secure variants of Dynamic Source Routing (DSR), like Ariadne [18], are vulnerable to network layer rushing attacks that only require a single insider node [19]. It causes any route of two or more hops to be routed through the attacking node by

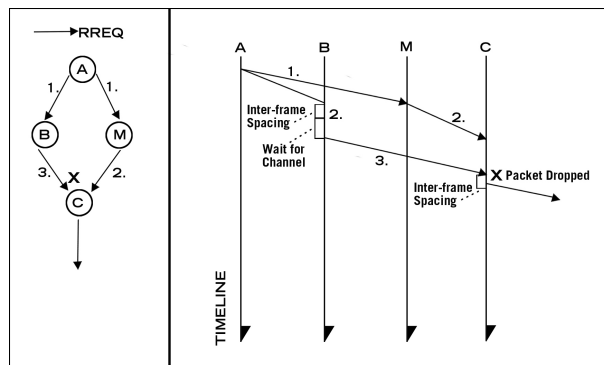


Figure 3: Rushing Attack: 1. RREQ broadcast from node A received by nodes B and malicious node M. 2. Node B waits for Inter-frame spacing while node M rushes RREQ by broadcasting immediately. 3. After waiting for a clear channel node B broadcasts RREQ. This is dropped by node C because it has already received this RREQ from node M. Any route established will now pass through the malicious node.

exploiting the property of AODV which requires RREQ messages to be only rebroadcast once by each node (to avoid broadcast storms). The attacking node broadcasts a rushed RREQ which will reach other nodes before legitimate RREQs, causing the legitimate RREQs to be discarded when they arrive later. As a result, the initiator node will be unable to find any route longer than two hops that does not include the attacker. This attack is not difficult for a single authenticated attacker to execute because on-demand routing protocols delay RREQ messages in two ways. Messages are delayed at the MAC layer, by exponential backoff and by interframe spacings mandated by the IEEE 802.11 protocol (see Figure 3), the typical wireless technology assumed by most, if not all, MANET routing protocols. Furthermore, routing protocols often specify a delay between receiving and rebroadcasting a RREQ to prevent collisions [19].

Another strategy for rushing RREQ messages is for the attacker to broadcast its RREQ at a higher power level than the other nodes, thereby transmitting over a greater distance and skipping hops to increase the chance of its RREQ reaching the destination first. However, the RREP will not be able to return on the route taken by the RREQ. The danger is that this will block legitimate RREQs because nodes along the path have already rebroadcast the malicious RREQ. Authentication is not required to conduct this attack, the RREQ can be replayed as is, making the malicious node entirely transparent.

AODV has a mechanism for dealing with unidirectional links that fits this scenario [2]. When a node receives a RREQ from a neighbour but notices the corresponding RREP is lost on the return path, it blacklists RREQs from that node for a BLACKLIST_TIMEOUT_PERIOD. This is to prevent RREQs propagating along bi-directional links from being blocked by RREQs propagating along undesirable uni-directional links.

3.3. Blackhole Attacks

AODV is vulnerable to the classic “Blackhole” attack defined for on-demand networks [20]. A composite and protocol non-compliant attack, the malicious node replies to every route request with a route reply, then drops the data packets. Grey Hole attacks are designed to defeat trust-based mechanisms. The attacking node does not act maliciously for an initial period to gain trust before beginning to misbehave [20]. SAODV has no mechanisms to deal with these kinds of routing protocol compliant dropping attacks by authenticated nodes. This attack can be crippling to the network when combined with other route attraction attacks. If the malicious node both rushes RREQs and broadcasts them at a higher power, unless the network has a large amount of redundancy, it is possible that no other route can be found that does not pass through the malicious node.

3.4. Resource Depletion Attacks

SAODV is still vulnerable to DoS in the form of resource depletion attacks. Authenticated nodes can simply send out unnecessary routing control messages, which can congest channels, use up battery power, and fill storage space. Nodes could also establish communications with other nodes and simply send unnecessary or fake data repeatedly. Such attacks pose substantial threats to critical or military applications, and may be hard to distinguish from normal communications.

3.4.1. Gratuitous RREQ Messages

As RREQ messages are broadcast throughout the entire network, persistent flooding of unnecessary RREQs by an authenticated nodes can dangerously consume network resources [21].

3.4.2. Gratuitous RERR Messages

Authenticated nodes may send out arbitrary RERR messages for any node whose existence they are aware. In AODV, when a node detects a broken link with its neighbours, it checks its routing table to see which routes are invalidated. The precursor nodes on this route are notified with RERR messages. According to RFC3561 [2], these RERR messages can be broadcast (if there are many precursor nodes) or unicast to each precursor. A node will only process a RERR message it received if that neighbour is part of an active route in its routing table. Therefore, RERR messages are not flooded throughout the network. They are passed along the routes that they are deleting. This means a malicious node sending RERRs for a route that does not pass through it will be ignored by its neighbours nodes. This limits the damage in terms of route disruption or resource depletion that gratuitous RERRs can cause. The worst a malicious node can do is isolate itself from the network.

3.4.3. Gratuitous Data Flows

In SAODV, there is no mechanism to secure data flows. This means that even an authenticated node could inject data flows onto a route once established [22]. An insider node could set up purposeless data flows to any point in the network up to its maximum transmission bandwidth. Since legitimate/malicious data flows may be indistinguishable, this kind of attack is difficult to guard against.

3.5. Distributed Denial of Service (DDoS) Attacks

Many of the attacks discussed so far can be considered as Denial of Service attacks, and the effects of these can be compounded when faced with multiple attackers. Coordinated DoS attacks by multiple users working in collusion result in a serious threat, known as Distributed Denial of Service (DDoS). Firstly, massive DDoS that can take place on the World Wide Web is less of a concern since the attackers in MANETs need to be within physical broadcast range. But, multiple unauthenticated attackers could seriously affect MANET functionality. They could simply jam the MANET by transmitting continuously on the channels the MANET is using. Even if routing control messages are cryptographically protected, there is a processing cost in checking and dropping unauthenticated routing packets. This could clog the channel or exceed node processing power, causing legitimate messages to be lost.

3.6. Jellyfish Attacks

Jellyfish attacks work on MANETs that use protocols with congestion control techniques, such as the Transmission Control Protocol (TCP), in the transport layer [9]. Packets can be maliciously dropped, reordered, or delayed to adversely affect network throughput. Since these attacks do not violate the protocol, their detection and diagnosis is time consuming and difficult. The jellyfish re-

ordering attack targets TCP's use of cumulative acknowledgement. TCP assumes that reordering of packets during transmission is relatively rare, and a node simply places packets into a reordering buffer instead of a FIFO one before delivering them, causing constant retransmissions.

Jellyfish Periodic Dropping Attacks exploit a weakness in TCP which means that if packet losses occur periodically near the Retransmission Time Out (RTO) time-scale, then end-to-end throughput is almost reduced to zero. The attacking node must drop all packets for a small time duration (10s of milliseconds) once per RTO (about every second) forcing TCP to continuously enter the timeout state. In a Jellyfish Delay Variance Attack, an attacking node waits a random time before forwarding each TCP packet, thus increasing the delay variance. High delay variance will cause TCP to transmit packets in bursts because of its "self-clocking" mechanism, resulting in increased collisions and packet loss. In delay-based congestion protocols, such as, TCP Westwood and Vegas, high delay variance leads to mis-estimation of bandwidth availability and excessively high RTO values [9].

3.7. MAC Layer Misbehaviour

The SAODV protocol was not designed to withstand DoS attacks [10]. While some proposed secure routing schemes for MANETs claim to prevent DoS, these may not be addressing the problem at the right layer as DoS is considerably simpler and more effective to achieve through MAC layer attacks [8].

The effects of MAC layer attacks on the network layer have been investigated for scenarios where the IEEE 802.11 protocol is used to provide link layer connectivity in MANETs [23]. Like AODV, IEEE 802.11 has been designed under the assumption that all nodes are protocol-compliant. The exponential backoff strategy used by IEEE 802.11 to handle contention for the channel can be an attack vector for link layer DoS [23]. The IEEE 802.11 standard specifies that all nodes that wish to transmit on a channel wait for a backoff period randomly selected from a certain interval. A node that limits its backoff interval can continuously capture the channel and starve other nodes from being able to transmit.

The facility for reserving the channel using the exchange of Request-to-Send and Clear-to-Send (RTS/CTS) in IEEE 802.11 is also open to abuse [23]. The RTS/CTS messages contain a field specifying the duration of time after the current frame needed by the node to complete the transmission of the frame. Any node within transmission range of the sending node or the receiving node that receives a RTS/CTS will update its network allocation vector (NAV) which specifies the duration of time for the node to defer transmission.

The results presented in [23] show the effects on the network layer of MAC layer misbehaviour. Suppose node A reduces its backoff time to continuously capture the channel. Now, neighboring node B will be unable to send

its regular HELLO messages. After missing a certain number of HELLO messages (usually 2), node A's neighbour C will mark the link to/from B as broken and inform their neighbours of this using RERR messages. This means that routes through B are now invalid, and this increases the probability that any workable route that used B will now go through node A. In this way, misbehaving nodes attract routes to themselves by blocking the transmissions of neighboring nodes. In the case of an intermittently misbehaving node, data packets will now be routed through it giving it the opportunity to sniff packets, drop packets or perform protocol compliant Jellyfish attacks. In the case of a continuously misbehaving node, its neighbours will be unsuccessful in their attempts to route packets through it as the channel is constantly blocked, and this will cause packets to back up at these nodes until their buffers overflow and packets are dropped.

None of the secure AODV variants offer MAC layer authentication leaving this attack avenue open to even unauthenticated nodes. Incorporating a secure MAC protocol can help alleviate this. However, an attacker attempting to jam the channel would only have to introduce a single bit error into a packet to corrupt it. In extreme attack environments, even more sophisticated frequency hopping techniques will be needed to try to circumvent jamming.

4. Solutions to threats against SAODV

This section provides a review of solutions that have been proposed to counter threats against SAODV. These are organised by category of threat. The categories are: (1) Wormhole attacks; (2) Rushing attacks; (3) Selfish node attacks; (4) Resource depletion attacks, and (5) Multiple attacks (combinations of the previous four categories).

4.1. Wormhole Attacks

4.1.1. Packet leashes

A packet leash is a technique to prevent wormholes by restricting the maximum allowed transmission distance of a packet [24]. These may be geographic or temporal. A geographic packet leash requires nodes to know their own location, and incorporate this information (cryptographically protected) into packets. This allows the distance from sender to receiver to be established. Temporal packet leashes require nodes to have tightly synchronized clocks. The packet creation time is included with the packet (encrypted), and this allows the receiver to estimate the distance a packet has traveled by examining the time the packet has been in transit. Of course, there is nothing to prevent a malicious authenticated node falsifying time stamps to make transit times appear shorter than they actually are.

4.1.2. Directional antennas

Another proposal for combating wormhole attacks uses directional antennas to implement a strict neighbour dis-

covery protocol [25]. Nodes are oriented with several directional zones of transmission, and the direction of transmission is included in neighbour discovery packets. When receiving transmissions, the directional antennas allow a node to establish the zone from which a transmission is received. Nodes only accept each other as neighbours when the direction of transmission is the opposite of the directional zone in which the transmission is received. Packets received from an unexpected direction are ignored. This means a wormhole attack or rushing attack can only be successful when the node impersonating a distant node as neighbour is directly between sender and receiver. This may be a useful solution for applications which already use directional antennae, otherwise the extra complexity associated with this solution may not be justifiable.

While the last two solutions may work to prevent wormholes, it is questionable whether they are worth the effort in terms of complexity and resources expended when compared to the danger that wormholes represent. A wormhole by itself does not present a threat to the MANET. By providing a shortcut across the network, the attackers are in fact providing a valuable service. The wormhole route attraction is only a threat when mixed with dropping of data packets or packet sniffing. Packet sniffing of sensitive data can be protected against with end-to-end encryption. However the danger is that by attracting enough routes, the sheer volume of encrypted data will facilitate the cracking of the encryption algorithm. While wormholes may allow an attacking node to attract routes to itself, the open wireless medium makes it difficult to prevent a well placed attacker from eavesdropping on transmissions. Dropping attacks may be more dangerous when used in conjunction with a wormhole, but if we have prevention mechanisms for dropping attacks in place, specific mechanisms for combating wormholes may not be necessary. The blocking of legitimate RREQs by RREQs rushed through the wormhole is another issue but, as we will see, other mechanisms could mitigate this threat.

4.2. Rushing Attacks

Blackhole attacks in SAODV can be defeated by flow triggered multipath routing. But, when used in conjunction with wormholes and rushing attacks, blackholes or the dropping of RREPs present a very real danger to both AODV and SAODV. As RREQs are only rebroadcast by receiving nodes once, a node that rushes or tunnels RREQs could seriously degrade the function of the on-demand algorithm by causing legitimate RREQs to be dropped. Flow monitoring and multipath routing can be used to find backup routes, if they exist, to alleviate this. But, there will be situations where all routes to the destination can be blocked or attracted towards the attacker.

4.2.1. Randomising RREQ rebroadcast

A solution for countering rushing attacks has been proposed in [19]. To prevent rushing attacks through a node

using higher power transmissions or out of band links to skip nodes, a secure neighbour detection protocol ensures that nodes will not forward RREQs from nodes that are not their neighbours. To deal with rushing attacks that employ the cutting of backoff times involves removal of the mechanism that only forwards the first RREQ received. Instead, the RREQ to be forwarded is selected randomly, meaning RREQs that arrive earlier (with low latency) are only slightly more likely to be forwarded. In response to a rushing attack preventing a route being found, re-initiating route discovery allows another chance at finding a valid route. Unfortunately, this would also mean that even if there is no threat, there would still be a chance of taking a non-optimal route, leading to some inefficiency. This paper also proposes a secure neighbour detection protocol based on tight monitoring of transmission times to set a maximum transmission distance and to prevent neighbour impersonation by tunneling and rebroadcasting packets (as in a Wormhole attack)[19].

4.3. Selfish Node Attacks

The selfish node problem becomes relevant in resource constrained public ad-hoc networks, where there needs to be an incentive for nodes to participate in routing. A secure incentive protocol is proposed in [26]. This is based on the idea of rewarding nodes for packets they have forwarded by giving them credits charged to the packet source and destination. The idea is that a node that uses network resources excessively without contributing by routing packets for other nodes will soon run out of credit. While this scheme does limit the damage to the network that could be done by selfish/malicious nodes, the credit system opens up new possibilities for exploitation. The scheme presented in [20] is secured using purpose-built devices that contain a tamper proof “secure module”, similar to a SIM card to manage credits and to cryptographically stamp secure tokens to be sent as payment with packets. It also incurs additional routing overhead as the payment terms are negotiated by nodes in advance.

In future public ad-hoc networks of mobile devices, such a payment scheme would be useful to achieve fairness in bandwidth usage among users. One potential problem could arise in the outlier nodes. A node that is not on the routes selected by the other nodes may not have the opportunity to forward packets to earn credit and would soon become “bankrupt”. In MANETs used for military applications or disaster relief communication, cooperation among nodes is mandatory and there is no need for a measure to ensure cooperation. The real concern, however, is resource depletion attacks and the incentive schemes can provide some defense against these attacks, e.g. the malicious node will soon use up all its credits if it attempts to flood the network with traffic. However, the outlier node problem makes this approach unsuitable for applications where resources are critical.

In public ad-hoc networks, it may be difficult to provide “tamper proof” cryptographic modules. Limiting partic-

ipation to specific purpose built devices may also not be desirable. A solution specific to SAODV has been proposed where the system of credits is maintained in neighbour nodes [27]. A node keeps a credit tally for each of its neighbours, which represents the probability of dropping or forwarding a RREQ by a neighbour node. While dealing with RREQ flooding attacks, this scheme is still vulnerable to resource depletion attacks in the form of gratuitous long lived data flows. This paper also proposes techniques to mitigate the computational load of SAODV’s use of encryption for routing control messages. It recommends disallowing RREPs by intermediate nodes with a route to the destination because of the computationally heavy double signatures. A trust-based mechanism allows the use of unauthenticated routing messages on paths between trusted nodes. This trust is established by observing correct transmissions of RREQs and RREPs. This criteria is not perfect, since an attacking node could still tamper with unencrypted messages.

4.4. Resource Depletion Attacks

Relatively fewer research efforts have addressed the problem of DoS through resource depletion by insider nodes and this presents a considerable threat in applications such as wireless sensor networks or battlefield communications. We need to guard against a captured and subverted node being used to drain the network of resources prematurely.

4.4.1. Gratuitous RREQ Messages

Gratuitous RREQ messages are resource intensive as they are broadcast through the whole network and they constitute a resource depletion attack vector. A soft state solution to the AODV/SAODV RREQ flooding problem has been proposed [21]. RFC3561 [2] specifies the RREQ-RATE-LIMIT parameter with a value of 10, but a malicious insider node could disable or increase this value, congesting its neighbourhood with RREQ messages. The RREQ-ACCEPT-LIMIT parameter limits the number of RREQs per second that a node will accept and forward. Also, the parameter RREQ-BLACKLIST-LIMIT is the threshold of RREQs per second that a neighbour node can send before being blacklisted by the listening node. The listening node will not accept RREQs from the blacklisted node for a period defined by BLACKLIST-TIMEOUT, which doubles each time a node is blacklisted. This is a simple solution to a limited problem and should be included in implementations of AODV/SAODV to prevent RREQ flooding attacks.

4.4.2. Combating Gratuitous Data Flows

TIARA, short for techniques for intrusion resistant ad-hoc routing algorithms, is a set of techniques for increasing robustness to intrusion attacks that are independent of the routing algorithm [22]. These include Flow Based Route Access Control (FRAC), multipath routing, flow monitoring, fast authentication, and network resource allocation.

They are general solutions that are not specific in their implementation nor tied to any particular on-demand routing protocol.

By only authenticating routing protocol packets, the SAODV protocol is open to resource depletion attacks where data flows are injected into an authenticated route. TIARA includes techniques for preventing gratuitous data flows. FRAC can be understood as a distributed firewall and is a first line of defense against unauthorized resource depletion attacks. Each node must maintain an access control list of allowed flows. This requires modifying the underlying protocol such that the routing table is indexed by flow identifier and routing decisions are made using this identifier.

TIARA increases the efficiency of FRAC by using lightweight fast authentication. The path label is placed in a node specific secret location within the flow’s data packets. The location of the path label is distributed to each node along the path as it is established by the routing algorithm. To prevent replay attacks sequence numbers are also added to secret locations in data packets in the same manner.

TIARA recognizes the danger of resource depletion attacks by one or more authorized nodes to battery powered networks. Referral Based Resource Allocation is a technique to combat these attacks. Routing nodes set out an initial threshold of network resources that may be used by a flow. Flows can exceed this threshold by receiving a certain number of “referrals” from trusted nodes that the request for network resources is reasonable. Determining what constitutes reasonable resource use is very task specific and difficult to generalize.

4.5. Multiple Attacks

4.5.1. An Intrusion Detection System

A soft state scheme specifically designed for SAODV to solve the dropping control packet attacks and wormhole attacks is presented in [17]. To combat dropping attacks, nodes within listening range of sending and receiving nodes keep track of control packets that are sent by one node but not forwarded by the next. Once the number of dropped packets reaches a threshold level, nodes in range will send a route elimination packet (REP) containing the malicious/selfish node’s identifier as well as the sending node’s identifier and signature. Nodes receiving a REP informing them of a blacklisted node will break their routing links through that node, isolating it from the network. To combat wormhole attacks, each node keeps a cache of all routes to a destination with the hop count information that is derived from RREPs. This allows nodes to compare the best route with the next best route, and if they differ by a sufficiently large ratio, it assumes a wormhole attack is taking place. The node that sent the RREP is then blacklisted and routes through this node are avoided where possible. The effectiveness of this solution needs to be tested, as in certain situations, the shortest route may have a hop count considerably smaller than the next shortest, leading to inefficient routing under this scheme.

On the whole, blacklisting of nodes that drop packets seems a good idea, but the difficulty lies in determining the threshold value. This solution is complicated because it is hard to tell the difference between natural channel interference and a node transmitting corrupted packets deliberately or a node unable to transmit because of a neighboring selfish node hogging the channel. Although perhaps in all these cases, it would be justifiable to blacklist the node and find a new route. Since blacklisting a node means we need to reroute around it, it may be more efficient to adapt AODV to find back up routes. We also need to assess whether it is really necessary to identify the misbehaving node, as this requires all nodes to constantly monitor their neighbours. It could be more efficient to trigger rerouting at the sending node when throughput drops, provided there are transport layer acknowledgments from destination node.

4.5.2. A Self Healing Community

The idea of a localized self healing community of nodes to observe packet transmission and use redundant routes to bypass nodes that drop packets has been proposed [28]. It also incorporates the idea of using RREQ rate limits to defend against resource depletion attacks conducted by authenticated nodes continuously flooding RREQs [28]. The functioning of a self healing community of nodes is outlined in Figure 4. Suppose Node B is on the route between nodes A and C. Any other nodes within transmission range of both 'a' and 'c' are the redundant nodes that could take over from 'b' (the localized self healing community). The redundant node 'd' overhears a packet forwarded to B but not passed on to C within a specified window, this node will then transmit the data packet to C itself. A similar concept, 'Witness Aided Routing' also proposes the use of redundant nodes to back up packet transmission [29].

Another option is to re-route through the redundant node, and this is done when a RREP is dropped during the route discovery phase (RREQs are always rebroadcast). To do this, a new field is added to the RREQ message which includes the forwarding node's immediate upstream neighbour. This is then recorded by nodes who receive the RREQ. If any node overhears a node receiving a RREP and not forwarding it, it then uses the recorded upstream neighbour information to send the RREP instead, thereby attracting the route to itself. This scheme has the advantage of healing every dropped packet (provided there is sufficient redundancy) leading to lower end-to-end delivery failures in the face of droppings attacks compared an intrusion detection scheme [17] which only blacklists a node after a threshold of dropped packets is reached.

Localized self healing communities of nodes is a promising idea that increases a network's robustness against malicious nodes and channel interference. It removes the need for blacklisting of nodes because any disturbance of routing paths is automatically accommodated for and, as such, increases resistance to MAC layer attacks. This mecha-

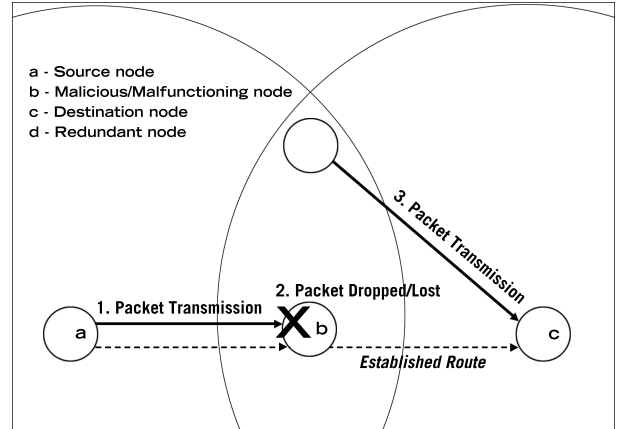


Figure 4: 1.Data packet transmitted along established route, redundant node overhears and buffers transmission. 2. Packet is dropped, redundant node does not hear transmission to destination. 3. After wait period, redundant node steps in to complete transmission

nism combats Blackhole attacks but does not defeat tunneling/wormhole or jellyfish attacks.

4.5.3. Multipath Routing (TIARA)

The security schemes that prevent dropping attacks have weaknesses. Firstly, it may be difficult to differentiate between regular wireless interference and malicious packet dropping, especially if the packets are only dropped periodically. Secondly, Jellyfish attacks at the transport layer can disrupt data flow. Instead of distributed monitoring for dropped packets, TIARA utilizes transport layer acknowledgements to monitor throughput. Loss of throughput at the transport layer could trigger re-routing at the network layer. Monitoring transport layer throughput is achieved by flow status messages. Path failure would be triggered if no flow message had been received in a specified time or if throughput drops below a certain threshold. AODV can be easily adapted so a destination will send multiple RREPs, allowing multiple or backup paths to be found if multiple paths exist. Multipath extensions to AODV have been proposed and can be utilized for this purpose [30]. Coupled with the ability to monitor throughput, and provided there is sufficient redundancy, multipath routing could help defeat all flow disruption attacks identified in this paper, as demonstrated by [31]. TIARA advocates modifying routing tables in on-demand routing protocols to be able to support next-hop information for multiple paths per flow [22]. Source-initiated flow routing using path labels is required to allow the sending node to choose from the multiple paths to the destination (see Figure 5).

5. Key Findings

Table 3 summaries the different attacks on AODV surveyed in this paper and the schemes that have been proposed to addressed these attacks. We have noted in the

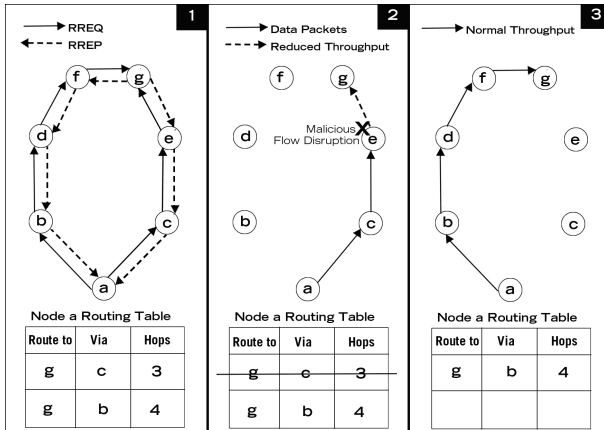


Figure 5: Tiara Multipath Routing

summary table the different degrees at which attack risks are reduced.

SAODV provides a cryptographic solution to secure routing control messages. However, SAODV still has vulnerabilities to attacks by captured or subverted insider nodes and even to unauthenticated attackers. Blackhole attacks block data transmission by dropping data packets while Jellyfish attacks achieve this by manipulating the transport layer protocol. MAC layer misbehavior can jam channels and block transmission by surrounding nodes. Wormholes and Rushing attacks allow attraction of routes to increase the effectiveness of data interception or Blackhole/Jellyfish attacks.

Solutions to wormhole attacks have been proposed in the form of packet leashes and secure neighbour protocols in the form of directional antennas. Further investigation is needed to understand the best way of combating these attacks since the wormholes are only a danger when used in conjunction with eavesdropping and dropping attacks. Rushing attacks and wormholes can block legitimate RREQs. One proposal combats this by randomizing the RREQ a node rebroadcasts.

Credit-based incentive schemes have been proposed that limit node resource usage, either in a distributed fashion or using tamper-proof cryptographic modules. Another proposal uses neighbour monitoring of RREQs to blacklist nodes that flood RREQs excessively.

Distributed intrusion detection that blacklists nodes which drop routing packets is complicated. There are many difficulties in differentiating malicious behavior from ordinary packet loss. Furthermore, monitoring neighbours may use a substantial amount of resources.

Self healing communities increase MANETs' robustness to jamming, dropping attacks, and channel interference without any traffic overhead. While not offering a complete solution, this approach has the potential for inclusion in MANET protocols, provided the network has sufficient redundancy. While monitoring neighbours is a non-trivial task (similar to distributed intrusion detection), self healing communities are more efficient in that

they instantly reroute around a malicious node. Intrusion detection requires propagation of the blacklist packets followed by repetition of route creation.

TIARA stands out among the proposals surveyed in this paper in that it attempts to find solutions to all the vulnerabilities in SAODV that we have identified. DoS through resource depletion seems to be an area that needs more attention. Generalization is difficult and solutions are highly domain specific. TIARA sets thresholds for network resource usage and uses distributed decision making in the form of referral-based resource allocation by trusted nodes to allow usage above this. This decision making could also be conducted from a centralized node. The criteria to be used will be domain dependent. TIARA also includes a form of distributed firewall, flow based route access control as a first line of defense against resource depletion, and proposes a fast authentication method to reduce the computational load required to implement this access control.

TIARA also proposes modifying the AODV protocol to find and remember multiple paths to the destination. A flow monitoring mechanism is added (although this could be achieved through monitoring feedback from the transport layer protocol). The idea is that a drop in throughput to below a threshold triggers the source to reroute the flow along a different path.

6. Future Research

We hope that future research on secure protocols will take a holistic approach to MANET security, by considering a spectrum of vulnerabilities from signal jamming to sophisticated attacks by authenticated nodes. It must be recognized that there is no "silver bullet" to MANET security and that a combination of different mechanisms will give better results. Some of the ideas surveyed in this paper should be considered for inclusion in such protocols, and we have identified three potential areas for future research.

Firstly, AODV's intrinsic ability find multiple routes should be utilized when throughput on a route diminishes. As wormholes and rushing attacks can attract all routes through the malicious nodes, blocking legitimate RREQs, a functional route may not be found. Randomizing RREQ rebroadcasts reduces this vulnerability but adds some inefficiency. We suggest another possible approach, triggering the rebroadcast of multiple RREQs per node when a functioning path cannot be found. This can be done but adding a flag to the RREQ message that causes nodes receiving the RREQ to rebroadcast one or more RREQ messages that may come after. This may result in some congestion temporarily but increases the possibility that a route around the wormhole can be found.

Secondly, determining how to allocate bandwidth in a MANET with scarce resources, in the presence of resource depletion attacks, is an open question that can be

Table 3: Summary of Attacks and Proposed Solutions

Attack Types	Proposed solutions								
	Packet Leashes	Directional Antennae	Incentive Schemes	RREQ Rate Limits	Randomising RREQs	TIARA - Flow Based Access Control	TIARA - Multipath Routing	Self Healing Communities	Blacklist/Intrusion Detection Schemes
Dropping Route Control Packets								•	•
Wormholes	•	•							
Rushing Attacks					•				
Blackhole Attacks			◦				•		
Selfish Nodes			•						
Resource Depletion			•	•		•			
Jellyfish Attacks							•		
MAC Layer Misbehaviour							•	•	◦

• Considerably Reduces Risk; ◦ Slightly Reduces Risk

a promising area for future research. RREQ rate limits are easily implemented and provide a first line of defense against resource depletion. Flow-based access control can provide another line of defense, while referral-based resource allocation needs to be tuned for specific MANETs.

Thirdly, the concept of Self Healing Communities or Witness Aided Routing can increase MANETs' robustness in many different situations without creating overhead. This idea should be developed for inclusion into secure protocols.

7. Conclusion

This paper has chosen AODV and SAODV as representative protocols to illustrate the scope of security vulnerabilities in MANET protocols. AODV uses unauthenticated routing control messages and has no mechanism for dealing with malicious manipulation of these messages. We analyze possible attacks on AODV with respect to attack vectors and attack goals. We then use the vulnerability profile of SAODV to examine proposed extensions that seek to combat these vulnerabilities. Possible attacks by both insider nodes and un-authenticated nodes are identified. Malicious nodes can prevent route creation, attract routes, disrupt data flows, or use up network resources. There have been many proposals for combating these weaknesses. Some present small alterations to the

routing protocol to combat specific weaknesses, while others present reactive soft-state security schemes that monitor the network for misbehavior.

Proposals to alter AODV include the imposition of RREQ limits to prevent gratuitous flooding and randomization of RREQ forwarding to counter rushing attacks. Two schemes use neighbour monitoring to identify nodes which drop routing control packets. The first is an intrusion detection system that blacklists misbehaving nodes, while the second is a self healing community where redundant nodes step in to retransmit dropped packets. Proposals to combat Wormhole attacks include the use of directional antennae to establish the direction of transmission and temporal/geographic packet leashes to limit the time/distance a packet may travel. We also examined TIARA, a comprehensive system of techniques to combat multiple weaknesses in ad-hoc routing protocols, including altering on-demand algorithms to remember multiple routes and using flow monitoring to trigger rerouting. We found the TIARA approach to be very attractive with a good potential to be included in MANET protocols.

References

- [1] H. Yang, H. Luo, F. Ye, S. Lu, L. Zhang, Security in mobile ad hoc networks: Challenges and solutions, *IEEE Wireless Communications* 11 (1) (2004) 38–47.
- [2] C. Perkins, E. Belding-Royer, S. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561.

- [3] M. Guerrero-Zapata, Secure ad hoc on-demand distance vector routing, *ACM SIGMOBILE Mobile Computing and Communications Review* 6 (3) (2002) 106–107.
- [4] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, E. M. Belding-Royer, A Secure Routing Protocol for Ad Hoc Networks, in: *Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP)*, Paris, France, 2002, pp. 78–89.
- [5] Q. Li, Y.-C. Hu, M. Zhao, A. Perrig, J. Walker, W. Trappe, Sear: a secure efficient ad hoc on demand routing protocol for wireless networks, in: *Proceedings of the 2008 ACM symposium on Information, computer and communications security, ASIACCS '08*, ACM, New York, NY, USA, 2008, pp. 201–204.
- [6] M. Mohammadzadeh, A. Movaghar, S. M. Safi, SEAODV: secure efficient AODV routing protocol for MANETs networks, in: *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human (ICIS)*, Seoul, Korea, 2009, pp. 940–944.
- [7] P. Ning, K. Sun, How to misuse AODV: a case study of insider attacks against mobile ad-hoc routing protocols, in: *Proceedings of the IEEE Systems, Man and Cybernetics Society Information Assurance Workshop (IAW)*, West Point, New York, USA, 2003, pp. 60–67.
- [8] M. Guerrero-Zapata, N. Asokan, Securing Ad hoc Routing Protocols, in: *Proceedings of the 2002 ACM Workshop on Wireless Security (WiSe 2002)*, Atlanta, Georgia, USA, 2002, pp. 1–10.
- [9] I. Aad, J.-P. Hubaux, E. W. Knightly, Denial of service resilience in ad hoc networks, in: *Proceedings of the 10th annual international conference on Mobile computing and networking (MobiCom)*, Philadelphia, PA, USA, 2004, pp. 202–215.
- [10] M. Guerrero-Zapata, Re: [manet] one way hash in SAODV, IETF MANET Working Group Discussions (8 January 2003). URL <http://www.ietf.org/mail-archive/web/manet/current/msg01443.html>
- [11] A. Perrig, R. Canetti, J. D. Tygar, D. Song, The TESLA Broadcast Authentication Protocol, *RSA CryptoBytes* 5.
- [12] R. Akbani, T. Korkmaz, G. Raju, HEAP: A packet authentication scheme for mobile ad hoc networks, *Ad Hoc Networks* 6 (7) (2008) 1134 – 1150. doi:10.1016/j.adhoc.2007.11.002.
- [13] Y.-C. Hu, D. Johnson, A. Perrig, SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks, in: *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, Callicoon, NY, USA, 2002, pp. 3–13.
- [14] C.-Y. Tseng, P. Balasubramanyam, C. Ko, R. Limprasittiporn, J. Rowe, K. Levitt, A specification-based intrusion detection system for AODV, in: *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks, SASN '03*, ACM, New York, NY, USA, 2003, pp. 125–134.
- [15] J. Grnkvist, A. Hansson, M. Skld, Evaluation of a specification-based intrusion detection system for aodv.
- [16] G. Vigna, S. Gwalani, K. Srinivasan, E. M. Belding-royer, R. A. Kemmerer, An intrusion detection tool for aodv-based ad hoc wireless networks, in: *20th Annual Computer Security Applications Conference*, 2004, pp. 16–27.
- [17] R. Chunxiao Chigan; Bandaru, Secure node misbehaviors in mobile ad hoc networks, in: *Vehicular Technology Conference, 2004., VTC2004-Fall. 2004 IEEE 60th, IEEE Computer Society, Washington, DC, USA, 2004*, pp. 4730–4734. doi:10.1109/VETEFCF.2004.1404990.
- [18] Y.-C. Hu, A. Perrig, D. B. Johnson, Ariadne: a secure on-demand routing protocol for ad hoc networks, *Wireless Networks* 11 (1-2) (2005) 21–38.
- [19] Y.-C. Hu, A. Perrig, D. B. Johnson, Rushing attacks and defense in wireless ad hoc network routing protocols, in: *Proceedings of the 2nd ACM workshop on Wireless security, WiSe '03*, ACM, New York, NY, USA, 2003, pp. 30–40. doi:<http://doi.acm.org/10.1145/941311.941317>.
- [20] P. Agrawal, R. K. Ghosh, S. K. Das, Cooperative black and gray hole attacks in mobile ad hoc networks, in: *Proceedings of the 2nd international conference on Ubiquitous information management and communication (ICUIMC)*, Suwon, Korea, 2008, pp. 310–314.
- [21] D. Gada, R. Gogri, P. Rathod, Z. Dedhia, N. Mody, S. Sanyal, A. Abraham, A distributed security scheme for ad hoc networks, *Crossroads* 11 (2004) 5–5.
- [22] R. Ramanujan, A. Ahamad, J. Bonney, R. Hagelstrom, K. Thurber, Techniques for intrusion-resistant ad hoc routing algorithms (TIARA), in: *Proceedings of the 21st Century Military Communications Conference (MILCOM)*, Vol. 2, Los Angeles, CA , USA, 2000, pp. 660–664.
- [23] L. Guang, C. Assi, Vulnerabilities of ad-hoc network routing protocols to mac misbehavior, in: *Proceedings of the IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob)*, Montreal, Canada, 2005, pp. 146–153.
- [24] Y.-C. Hu, Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks, in: *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications (INFOCOM)*, San Francisco, CA, USA, 2003, pp. 1976–1986.
- [25] L. Hu, D. Evans, Using Directional Antennas to Prevent Wormhole Attacks, in: *Proceedings of the Network and Distributed System Security Symposium*, San Diego, CA, USA, 2004.
- [26] Y. Zhang, W. Lou, W. Liu, Y. Fang, A secure incentive protocol for mobile ad hoc networks, *Wireless Networks* 13 (2007) 569–582.
- [27] F. De Rango, S. Marano, Trust-based saodv protocol with intrusion detection and incentive cooperation in manet, in: *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly, IWCMC '09*, ACM, New York, NY, USA, 2009, pp. 1443–1448.
- [28] J. Kong, X. Hong, Y. Yi, J.-S. Park, J. Liu, M. Gerla, A secure ad-hoc routing approach using localized self-healing communities, in: *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, Urbana-Champaign, IL, USA, 2005, pp. 254–265.
- [29] I. Aron, S. Gupta, A witness-aided routing protocol for mobile ad-hoc networks with unidirectional links, in: H. Leong, W.-C. Lee, B. Li, L. Yin (Eds.), *Mobile Data Access*, Vol. 1748 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 1999, pp. 24–33.
- [30] A. C. Valera, W. K. G. Seah, S. V. Rao, Improving Protocol Robustness in Ad Hoc Networks through Cooperative Packet Caching and Shortest Multipath Routing, *IEEE Transactions on Mobile Computing* 4 (5) (2005) 443–457.
- [31] W. Lou, W. Liu, Y. Zhang, Y. Fang, SPREAD: Improving network security by multipath routing in mobile ad hoc networks, *Wireless Networks* 15 (3) (2009) 279–294.