How to Misuse AODV: A Case Study of Insider Attacks against Mobile Ad-Hoc Routing Protocols [1,2]

Peng Ning and Kun Sun

Cyber Defense Laboratory, Computer Science Department, North Carolina State University, Raleigh, NC 27695-8207, USA

Abstract

This paper presents a systematic analysis of insider attacks against mobile ad-hoc routing protocols, using the Ad-hoc On-Demand Distance Vector (AODV) protocol as an example. It identifies a number of attack goals, and then studies how to achieve these goals through misuses of the routing messages. To facilitate the analysis, it classifies insider attacks into two categories: *atomic misuses* and *compound misuses*. Atomic misuses are performed by manipulating a single routing message, which cannot be further divided; compound misuses are composed of combinations of atomic misuses and possibly normal uses of the routing protocol. The analysis results in this paper reveal several classes of insider attacks, including *route disruption, route invasion, node isolation,* and *resource consumption*. Finally, this paper presents simulation results that validate and demonstrate the impact of these attacks.

Key words: mobile ad-hoc networks, insider attacks, AODV, routing protocol
PACS:

1 Introduction

Mobile Ad-hoc Networks (MANET) have attracted substantial research efforts recently, partially due to their appealing applications in infrastructureless sit-

Email addresses: pning@ncsu.edu, ksun3@ncsu.edu (Peng Ning and Kun Sun).

[1] A preliminary version of this paper appeared in the *Proceedings of 2003 IEEE Systems, Man and Cybernetics Society Information Assurance Workshop*, pages 60–67, June 18-20, 2003.

[2] This work is supported by the Army Research Office (ARO) under grant DAAD19-02-1-0219.

uations such as battle fields and disaster recovery operations. In MANET, each mobile node functions as both a host and a router. Mobile nodes are typically powered by batteries, and have less powerful computing resources than desktop computers. Moreover, the network topology is highly dynamic due to the movements of the nodes. These features introduce unique problems that do not appear in traditional, wired networks.

Among all the research issues, security in mobile ad-hoc routing protocols is particularly challenging due to the nature of wireless communication and the lack of infrastructure supports. Several efforts (e.g., Security-aware AODV [1], ARAN [2], SRP [3], Ariadne [4], SEAD [5], CONFIDANT [6], Watchdog and Pathrater [7]) are underway to provide security services in ad-hoc routing protocols. Most of the current security mechanisms (e.g., ARAN [2], Ariadne [4], SEAD [5]) are preventive approaches that depend on cryptography to ensure the security of the network. However, in a typical mobile ad-hoc network such as a battle field, mobile nodes are extremely vulnerable to capture or key compromise. Even if critical keying materials are protected by tamper-proof hardware, it is still difficult to ensure that the same hardware will not be misused by an attacker.

To ensure the security of the network, it is critical to develop security mechanisms that can survive malicious attacks from "insiders" who have access to the keying materials or full control of some nodes. In order to protect against insider attacks, it is necessary to understand how an insider can attack a wireless ad-hoc network. Several attacks (e.g., routing disruption attacks and resource consumption attacks [4, 5, 8]) have been discussed in the literature. However, insider attacks in general have not been thoroughly studied and verified.

In this paper, we adopt a systematic approach to study the insider attacks against mobile ad-hoc routing protocols. Our analysis scheme consists of two dimensions. The first is a set of all possible *atomic misuse actions* that an inside attacker may take to misuse a routing message; each atomic misuse action is an indivisible manipulation of one routing message. The second dimension is a set of *misuse goals* that an inside attacker may want to achieve. Our analysis is then to examine whether the attack goals can be achieved through combinations of atomic misuse actions.

To facilitate the analysis, we further classify misuses of a routing protocol into two categories: *atomic misuses* and *compound misuses*. Intuitively, an atomic misuse is performed by applying one atomic misuse action to a single routing message, which cannot be further divided. In contrast, compound misuses are composed of multiple atomic misuse actions and possibly normal uses of the routing protocol. Since atomic misuses are potentially "building blocks" of compound misuses, in this paper, we start our analysis with atomic misuses.

We then study compound misuses, especially those that can lead to more persistent and powerful impacts by repeating a single type of atomic misuses.

We pick the Ad-hoc On-Demand Distance Vector (AODV) protocol [9] as an example, performing our analysis from an attacker's perspective. Our analysis results indicate that though the AODV protocol may rely on other protocols (e.g., IPsec Authentication Header protocol [10]) to provide security services, one or multiple inside attackers who have access to the cryptographic keys can still successfully misuse the protocol messages to disrupt the network, invade routes, isolate valid nodes, or consume the network resources. To validate the analysis results, we have implemented all the misuses based on the AODV extension in ns2 [11], and evaluated the effectiveness of the misuses through simulations. Our experimental results indicate that all the misuses identified in this paper are real, and may affect the normal operation of MANET in various ways.

The contribution of this paper is two-fold. First, we develop a systematic approach to analyze insider attacks against MANET routing protocols. Besides the AODV protocol, this analysis scheme can be potentially applied to other protocols and help protocol designers develop secure and efficient routing protocols. Second, we identify a set of misuses against the AODV protocol, informing the practitioner the vulnerabilities and risks in AODV networks. Our implementation of these attacks and experimental results may also be used to support research on intrusion detection in MANET.

The rest of this paper is organized as follows. The next section briefly describes the AODV protocol. Section 3 describes our analysis scheme. Section 4 focuses on analyzing the atomic misuses of AODV routing messages. Section 5 discusses compound misuses. Section 6 presents the experimental validation of the misuses discovered in our analysis. Section 7 discusses related work, and Section 8 concludes this paper and points out some future research directions.

2 An Overview of The AODV Protocol

The AODV protocol [9] is an on-demand routing protocol, which initiates a route discovery process only when desired by an originating node. When an originating node wants to send data packets to a destination node but cannot find a route in its routing table, it broadcasts a Route Request (RREQ) message to its neighbors. Its neighbors then rebroadcast the RREQ message to their neighbors if they do not have a *fresh enough* route [3] to the destination

[3] A fresh enough route is a valid route to the destination node whose associated sequence number is equal to or greater than that contained in the RREQ message.

node. This process continues until the RREQ message reaches the destination node or an intermediate node that has a fresh enough route.

Every node has its own sequence number and RREQ ID [4]. AODV uses sequence numbers to guarantee that all routes are loop-free and contain the most recent routing information. The RREQ ID in conjunction with the originator IP address uniquely identifies a particular RREQ message. The destination node or an intermediate node only accepts the first copy of a RREQ message, and drops the duplicated copies of the same RREQ message.

After accepting a RREQ message, the destination or an intermediate node updates its *reverse route* to the originating node using the neighbor from which it receives the RREQ message. The reverse route will be used to send the corresponding Route Reply (RREP) message to the originating node. Meanwhile, it updates the sequence number of the originating node in its routing table to the maximum of the one in its routing table and the one in the RREQ message. When the originator or an intermediate node receives a RREP message, it updates its *forward route* to the destination node using the neighbor from which it receives the RREP message. It also updates the sequence number of the destination node in its routing table to the maximum of the one in its routing table and the one in the RREP message. A Route Reply Acknowledgment (RREP-ACK) message is used to acknowledge receipt of a RREP message. Though not required, AODV may utilize the HELLO message to maintain the local connectivity of a node.

Route maintenance is done with Route Error (RERR) messages. If a node detects a link break in an active route, it sends out a RERR message to its upstream neighbors that use it as the next hop in the broken route. When a node receives a RERR message from its neighbor, it further forwards the RERR message to its upstream neighbors.

AODV is a stateless protocol; the originating node or an intermediate node updates its routing table if it receives a RREP message, regardless of whether it has sent or forwarded a corresponding RREQ message before. If it cannot find the next hop in the reverse routing table, it simply drops the RREP message. Otherwise, it unicasts the RREP message to the next hop in the reverse route.

In general, a node may update the route entries in its routing table whenever it receives RREQ, RREP, or RERR messages from its neighbors.

[4] It is also known as flood ID in earlier versions of AODV specifications.

3 Analysis Scheme

As discussed earlier, we adopt a systematic way to analyze the insider attacks against the AODV protocol. We first identify all possible atomic misuse actions that an inside attacker may take to manipulate a routing message, and then a number of misuse goals that an inside attacker may want to achieve. We analyze the insider attacks against the AODV protocol from an attacker's perspective, studying how to use one or more atomic misuse actions to achieve these misuse goals.

The first dimension of our analysis scheme is a set of all possible atomic misuse actions. An atomic misuse action is an indivisible manipulation of one routing message. Specifically, we divide the atomic misuse actions into the following four categories:

- *Drop (DR).* The attacker simply drops the received routing message.
- *Modify and Forward (MF).* After receiving a routing message, the attacker modifies one or several fields in the message and then forwards the message to its neighbor(s) (via unicast or broadcast).
- *Forge Reply (FR).* The attacker sends a faked message in response to the received routing message. Forge Reply is mainly related to the misuse of RREP and RREP-ACK messages, which are in response to RREQ and RREP messages, respectively.
- *Active Forge (AF).* The attacker sends a faked routing message without being triggered by receipt of any routing message.

The above misuse actions are all that an attacker can do about a routing message. There may be other ways to classify or represent misuse actions, which are essentially one or a combination of the above atomic misuse actions. For example, a *replay attack*, by which an attacker buffers a previously received routing message and replays it later, is an "Active Forge (AF)" misuse action in the above classification, since the attacker sends a fake, unsolicited routing message (by replaying the previously seen message).

The second dimension of our analysis scheme is a set of misuse goals that inside attackers may want to achieve. As the reader may have realized, it is very difficult, if not entirely impossible, to list all possible misuse goals, since these misuse goals highly depend on the MANET applications that are not foreseen right now. In our analysis scheme, we list several typical misuse goals related to the routing layer below. Though we focus on these misuse goals in this paper, other misuse goals can be identified and analyzed in a similar way.

- *Route Disruption (RD).* Route disruption is to either break down an existing route or prevent a new route from being established.
- *Route Invasion (RI).* Route invasion attempts to add an attacking node into

a route between two communicating nodes. Once an attacker reaches this goal, he/she may launch other attacks (outside of the routing layer) such as selectively dropping and/or sniffing the data packets.

- *Node Isolation (NI).* Node isolation is to prevent a given node from communicating with any other node in the network. It differs from route disruption in that route disruption is targeted at a route with two given endpoints, while node isolation is aimed at breaking all possible routes to or from a given node.
- *Resource Consumption (RC).* Resource consumption is to consume the communication bandwidth in the network, computation resources, or storage space at individual nodes. Severe resource consumptions usually lead to denial of service attacks. Any action that causes a potential victim node to react consumes the node's resources. However, we are only concerned about the actions that force other valid nodes to consume much more resources than the attacking node.

When an attacker misuses a routing message, it may apply different atomic misuse actions to the same routing message. To achieve a misuse goal, in some scenarios, it is enough for an inside attacker to utilize one atomic misuse action on one routing message; however, in many other scenarios, the attacker has to perform a sequence of atomic misuse actions on several routing messages. To facilitate the analysis, we classify misuses of the AODV protocol into two categories: *atomic misuses* and *compound misuses*. Intuitively, an atomic misuse is performed by manipulating a single routing message, which cannot be further divided. In contrast, a compound misuse is composed of multiple atomic misuses and possibly normal uses of the routing protocol. It is easy to see that atomic misuses may be used as building blocks of compound misuses.

Some compound misuses are simple compositions of atomic misuses and normal uses of the routing protocol. With atomic misuses identified, these compound misuses can be derived automatically using vulnerability analysis tools [12–16]. We do not consider such compound misuses in this paper. However, when carefully aggregated together, some compositions of atomic misuses become more powerful attacks due to the changes in the number or the organization of the routing messages. For example, if an attacker periodically broadcasts RREQ messages with false information in the neighborhood of a victim node, the attacker can successfully prevent the victim node from receiving any messages. We will investigate such compound misuses after identifying the atomic ones.

It is easy to see that our analysis scheme is also applicable to other mobile ad-hoc routing protocols, possibly with slight modification. However, in this paper, we only focus on the AODV protocol, while considering the analysis of the other ad-hoc routing protocols or secure ad-hoc routing protocols as possible future work.

Since atomic misuses form the foundation of compound misuses, in the following, we first perform an analysis of atomic misuses of the AODV protocol, and then study how atomic misuses and normal routing messages may be combined to launch compound misuses.

4 Atomic Misuses of AODV

In our analysis, we use a simple naming scheme to identify atomic misuses, which combines routing message type and atomic misuse action. Specifically, each atomic misuse is named in the form of `MessageType_Action`, which means that an inside attacker applies the "`Action`" to a routing message of type "`MessageType`." For brevity, we use the abbreviations introduced in the previous section to represent atomic misuse actions. For example, `RREP_DR` represents that an attacker drops (DR) a RREP message. We also use names in the form of `MessageType_Action_Goal` to represent that an inside attacker attempts to achieve the "`Goal`" by applying the "`Action`" to a routing message of type "`MessageType`." For example, `RREP_DR_RD` represents that an attacker attempts to disrupt (RD) a route by dropping (DR) a RREP message.

In the following, we present our analysis results about the atomic misuses of RREQ, RREP, RERR, and RREP-ACK messages, respectively. For each type of routing messages, we first summarize our findings, and then discuss the atomic misuses in detail.

4.1 Atomic Misuses of RREQ Messages

An attacker may drop, modify and forward, or actively forge RREQ messages. However, the atomic misuse action *Forge Reply* is not applicable to RREQ messages, since RREQ messages are not used to reply to any other routing message in the AODV protocol.

Table 1 summarizes the atomic misuses of a RREQ message. Each cell in this table is marked "Yes", "No", or "Partial". "Yes" means that it is possible to achieve misuse goal in the column with the atomic misuse in the row. "No" means that it is not possible to achieve the misuse goal in the column with the atomic misuse in the row. "Partial", which is only used for the misuse goal Node Isolation, indicates that the atomic misuse in the row can either prevent the victim node from sending or receiving (but not both) data packets, thus partially achieving the misuse goal.

Table 1
Atomic Misuses of A RREQ Message and Achievable Misuse Goals.

Atomic Misuse	Route Disruption	Route Invasion	Node Isolation	Resource consumption
RREQ_DR	Yes	No	No	No
RREQ_MF	Yes	Yes	Partial	No
RREQ_AF	Yes	Yes	Partial	No

4.1.1 Atomic Misuse RREQ_DR

RREQ_DR refers to simply dropping the received RREQ message. If an attacker applies such misuses to all the RREQ messages it receives, it is equivalent to not having the attacking node in the network. An inside attacker may also selectively drop RREQ messages. Attackers that launch such misuses are in nature similar to the selfish nodes mentioned in [7]. If the attacking node is the only node between two parts of an ad-hoc network, it may selectively separate the nodes in these two parts, and achieve the goal of route disruption for a short period of time. It is easy to see that RREQ_DR cannot achieve the other three misuse goals.

4.1.2 Atomic Misuse RREQ_MF

RREQ_MF refers to an atomic misuse with which an inside attacker modifies one or several fields in a RREQ message that it just receives, and then broadcasts the modified RREQ message. Table 2 lists the RREQ message fields that an attacker may modify as well as the possible modifications. An attacker may also modify the IP addresses in the IP header. Let us first review the possible modifications of these fields and their consequences, and then discuss the atomic misuses that can be successfully launched.

Several fields have immediate security implications when modified. RREQ ID along with the originator IP address uniquely identifies a RREQ message; they indicate the freshness of a RREQ message. Since a node only accepts the first copy of a RREQ message, an attacker may use an increased RREQ ID along with the originator IP address to convince other nodes to accept the modified RREQ message.

To ensure loop freedom in AODV, after receiving a RREQ message, a node updates its reverse route table only if the originator sequence number field in the RREQ message is greater than that in its route table, or the originator sequence numbers are equal, but the hop count field in the RREQ message plus one is smaller than that in the route table. An inside attacker may also change these two fields to affect other nodes' route table.

Table 2
Possible Modifications of Fields in A RREQ Message.

RREQ Message Field	Modifications
Type	Change the message type.
RREQ ID	Increase it to make the faked RREQ message acceptable, or decrease it to make the RREQ message unacceptable.
Hop Count	Decrease it to update other nodes' reverse route tables, or increase it to suppress its update.
Destination IP Address	Replace it with another IP address.
Destination Sequence Number	Increase it to update other nodes' forward route tables, or decrease it to suppress its update.
Originator IP Address	Replace it with another IP address.
Originator Sequence Number	Increase it to update other nodes' reverse route tables, or decrease it to suppress its update.
Flags	Reverse the setting.

According to the AODV protocol, after receiving a RREQ message, the destination node updates its sequence number as the maximum of the one in the RREQ message and the one in its route table. An inside attacker may increase the destination sequence number in a faked RREQ message to update the destination node's sequence number.

When a node updates its route table, the next hop in the route entry is assigned as the node from which it receives the RREQ message, which is indicated by the source IP address in the IP packet that contains the routing message. An inside attacker may manipulate source IP address in the IP header to change the reverse route.

An inside attacker may also reverse the setting of the flag bits in one RREQ message. For example, by clearing the "D" flag, other intermediate nodes may be allowed to reply RREP messages, while the originating node only requests the RREP message from the destination node. However, the originating node cannot distinguish the RREP messages from the destination node or intermediate nodes.

In the following, we discuss the atomic misuses that can be launched by modifying and forwarding RREQ messages.

Atomic Misuse `RREQ_MF_RD`

If an attacking node is the only node connecting two parts of an ad-hoc network, the attacker can prevent a new route from being established by making one of the following modifications on a RREQ message it receives:

- Change the message type;
- Replace the destination IP address with another IP address;
- Replace the originator IP address with another IP address;
- Replace the source IP address (in the IP header) with another IP address.

Even if there exist other routes between the two communicating nodes, the attacking node still has a chance to prevent the new route from being established. Suppose node O broadcasts a RREQ message to establish a route to node D. After receiving the RREQ message, the attacking node may make the following modifications to the RREQ message:

- Replace the RREQ ID of node O with the RREQ ID of node D, and increases it by a small number;
- Interchange the originator IP address (node O) with the destination IP address (node D) in the RREQ message;
- Increment the destination sequence number by at least one, and then interchanges the originator sequence number with the destination sequence number;
- Fill the source IP address (in the IP header) with a non-existent IP address.

With these modifications, the attacking node pretends to forward a RREQ message initiated from node D to node O, whereas the original RREQ message is initiated from node O to node D. Neighbors of the attacking node will accept the faked RREQ message, since they have not received a RREQ message with such a RREQ ID from node D before. Because the faked RREQ message has a greater originator sequence number, these neighbors will update their next hop to node D as a non-existent node, which is indicated by the source IP address in the IP header. These neighbors will rebroadcast the faked RREQ message to their neighbors. When node D receives the faked RREQ message, it just drops the message since it notices that this message is originated from itself. When node O receives the faked RREQ message, it will update its reverse route table, since the originator sequence number (of node D) in the faked RREQ message is greater than the one in its route table. Node O will then update the next hop to node D as the neighbor from which it receives the faked RREQ message, and unicasts a RREP message to this neighbor. When the RREP message is unicasted along the reverse route, it will be lost due to the non-existent node in the reverse route.

Due to the broadcast of the legitimate RREQ message, node O may receive normal RREP messages. However, the route established by the faked RREQ

message will suppress the routes established by these normal RREP messages, since node D's sequence number in the faked RREQ message is greater than those in the normal RREP messages. If node O sends data packets along the route established by the faked RREQ message, all data packets will be dropped when they are sent to the non-existent node. When the upstream neighbor of the attacking node discovers the link failure, it will either send a RERR message back to node O, or start "local repair", which broadcasts a RREQ message to discover a route from itself to the destination node if the destination is not farther than the maximum number of repair hops.

Note that the reverse route (established by RREQ messages) and the forward route (established by RREP messages) are in one route table, though the route entries resulting from RREQ messages have shorter lifetime than those established by RREP messages. The route entries added by RREQ messages can be updated by RREP message, and vice versa.

Atomic Misuse `RREQ_MF_RI`

Let us first consider a scenario in which an inside attacker is in the transmission range of an originating node that initiates route discovery with a RREQ message. After receiving the RREQ message from the source node, the attacking node may modify the RREQ message as follows:

- Increase the originating node's RREQ ID by at least one;
- Increase the originator sequence number by at least one;
- Increase the destination sequence number by at least one.

After generating this faked RREQ message, the attacking node broadcasts it to its neighbors. These neighbors will accept this faked RREQ message due to the new RREQ ID. They will then update their next hop to the originating node as the attacking node, because the faked RREQ message has a greater originator sequence number than those in their route tables. They will also rebroadcast the faked RREQ message to their neighbors. When the originating node receives the faked RREQ message, it will drop the message, since this message appears to originate from itself. When the destination node receives the faked RREQ message, it will update its next hop to the originating node as the neighbor from which it receives the faked RREQ message, and then update its own sequence number to the destination sequence number in the RREQ message, which is greater than its current sequence number. After that, it will fill the updated sequence number into the destination sequence number in the RREP message. The destination node will then unicast the RREP message to the originating node along the reverse route, which includes the attacking node. Because this RREP message contains a greater destination sequence number than that in the originating node's route table, which may have been updated by other legitimated RREP messages, the originating node

will update the destination sequence number to the one in the RREP message, and set the attacker as the next hop to the destination node. As a result, the attacker succeeds in invading the route from the originating node to the destination node.

When the attacker is not in the transmission range of an originating node, that is, there exists at least one intermediate node between the attacking node and the originating node, the attacker cannot invade the route by modifying a RREQ message in the above way. When the attacking node broadcasts the faked RREQ message, all the neighbors will accept it and update the attacker as the next hop to the originating node. When the attacking node forwards the RREP message to a neighbor along the reverse route, this neighbor will just send the RREP message back to the attacking node. As a result, there will be a loop involving the attacking node and one of its neighbors. Nevertheless, an attacker can invade the route by sending two faked RREQ messages in a compound misuse.

Atomic Misuse `RREQ_MF_NI`

An attacker may launch a `RREQ_MF_NI` attack and prevent a victim node from receiving data packets from other nodes for a short period of time. Assume that the attacking node receives a RREQ message from the victim node. The attacker may then make the following modifications:

- Increase the RREQ ID by a small number;
- Replace the destination IP address with a non-existent IP address;
- Increase the originator sequence number by at least one;
- Set the source IP address (in the IP header) to a non-existent IP address.

The attacking node then broadcasts this modified message. When the neighbors of the attacker receive the faked RREQ message, they will update the next hop to the victim node to the source IP address in the IP header, since the faked RREQ message has a greater originator sequence number. Due to the non-existent destination IP address, this faked message can be broadcasted in the ad-hoc network until the lifetime of the packet ends. This message will then affect the reverse route (to the victim node) on all the nodes that rebroadcast this message. When other nodes want to send data packets to the victim node, they will use the reverse routes established by the faked RREQ message, which include the non-existent IP address (i.e., the source IP address in the IP header of the faked RREQ message). As a result, the data packets will be dropped when they are sent to the non-existent node. This atomic misuse can prevent a victim node from receiving data packets only for a short period of time, due to the local repair mechanism in the AODV protocol [9]. The other nodes will initiate another round of route discovery if they note that the data packets cannot be delivered successfully.

The victim node may still be able to send data packets to other nodes. Thus, this atomic misuse only partially achieves the node isolation goal.

Atomic Misuse `RREQ_MF_RC`

It is difficult for an attacker to consume much resource with one faked RREQ message. However, an attacker may still be able to introduce unnecessary broadcast messages into the network through a single `RREQ_MF_RC` misuse. Specifically, an attacker can modify an incoming RREQ message to make it appear to be fresh (by increasing the RREQ ID) so that it will be rebroadcast by the attacker's neighbors. To generate real impact on the network, the attacker needs to repeatedly apply `RREQ_MF_RC` misuses, and generate a broadcast message loop in the network. We will discuss such misuses in the context of compound misuses.

4.1.3 Atomic Misuses `RREQ_AF`

Both `RREQ_DR` and `RREQ_MF` must be triggered by an incoming RREQ message. In contrast, an inside attacker may perform a `RREQ_AF` misuse by actively forging a RREQ message without receiving a RREQ message. An inside attacker may need to collect some necessary information to forge RREQ messages (e.g., by overhearing the traffic). Theoretically, the attacker may forge any field in a RREQ message, generating the effects we just discussed.

Atomic Misuse `RREQ_AF_RD`

If there exists a route from an originating node to a destination node, an inside attacker can break down the route by broadcasting a faked RREQ message. In the faked RREQ message, the attacker pretends to rebroadcast a RREQ message initiated from the destination node to the originating node with a non-existent node as the source IP address in the IP header, just as described in `RREQ_MF_RD`. Due to the same reason described in `RREQ_MF_RD`, the originating node will update its route to go through the non-existent node to the destination node. As a result, the route will be broken.

Atomic Misuse `RREQ_AF_RI`

In an atomic misuse `RREQ_AF_RI`, an attacker may invade a route by generating a faked RREQ message actively. Assume the attacking node is in the transmission range of the originating node. It can generate a faked RREQ message as the one described in `RREQ_MF_RI`. That is, the faked RREQ message should have (1) a RREQ ID greater than the most recent RREQ ID in the RREQ message sent by the originating node, (2) a originator sequence number greater than the most recent sequence number of the originating node, and (3) a destination sequence number greater than the most recent sequence number of the

destination node. The attacking node then broadcasts this message, pretending to forward a RREQ message from the originating node to the destination node.

When the destination node receives this message, it will send back a RREP message according to the AODV protocol. This RREP message will reach the attacking node through the reverse route. The attacking node can then forward it to the originating node. After receiving the RREP message, the originating node will update the attacking node as the next hop to the destination node.

If the attacking node is not in the transmission range to the originating node (i.e., the victim node), it need have an existing route to the originating node so that the attacking node can forward the RREP message to it. Though this increases the requirement for the misuse, it still can achieve the misuse goal of route invasion if the condition is satisfied.

Atomic Misuse `RREQ_AF_NI`

This atomic misuse is similar to `RREQ_MF_NI`; the only difference is that the attacking node forges a RREQ message actively, without receiving a normal one. This requires that the attacking node gathers the information about the most recent RREQ ID and sequence number of the originating node that it tries to pretend. This atomic misuse can achieve the same effect as `RREQ_MF_NI`: an inside attacker can prevent a victim node from receiving data packets for a short period of time, but cannot completely prevent the victim node from sending data packets to other nodes, unless other misuses are also used.

Similar to `RREQ_MF`, an attacking node cannot consume much resource of the network with a single `RREQ_AF` misuse.

4.2 Atomic Misuses of RREP Messages

The premise of atomic misuses of RREP messages is that the inside attacker must already be in a reverse route involving a victim node, so that it can receive a RREP message, or send a forged RREP through some other nodes. Due to this restriction, most of the atomic misuses of RREP messages, including `RREP_DR` and `RREP_MF`, have limited impact.

Table 3 summarizes the atomic misuses of a RREP message and whether they can achieve the misuse goals. Similar to Table 1, the cells marked "Yes" (or "No") represent the atomic misuse in the row can (or cannot) achieve the misuse goal in the corresponding column.

Table 3
Atomic Misuses of A RREP Message and Achievable Misuse Goals.

Atomic Misuse	Route Disruption	Route Invasion	Node Isolation	Resource consumption
RREP_DR	Yes	No	No	No
RREP_MF	Yes	Yes	No	No
RREP_FR	Yes	Yes	No	No
RREP_AF	Yes	Yes	No	Yes

4.2.1 Atomic Misuse RREP_DR

It is trivial to analyze the atomic misuse RREP_DR. If an attacking node is in a strategic location such that all RREP messages have to pass through the attacking node, the attacking node can easily prevent the route from being established by dropping the RREP messages. However, when the originating node has at least one path to the destination node that does not include the attacking node, this misuse has little impact. It is obvious that RREP_DR misuses cannot invade a route, isolate a node, or consume noticeable network resources.

4.2.2 Atomic Misuse RREP_MF

RREP_MF refers to an atomic misuse with which an inside attacker modifies one or several fields in a RREP message that it just receives, and then forwards the modified RREP message. Table 4 lists the RREP message fields that an attacker may modify as well as the possible modifications. A RREP message contains some new fields that are not included in a RREQ message. The lifetime field determines the valid time of the route entry updated by the RREP message. When the RREP message is sent through an unreliable or unidirectional link, the sender may set the "A" flag, and the receiver of the RREP is expected to return a RREP-ACK message as an acknowledgment to the sender. The "prefix size" field is used to facilitate the route establishment in a subnet. An attacker may also modify the IP addresses in the IP header.

Atomic Misuse RREP_MF_RD

In a route discovery process, if the only RREP message passes through an inside attacker, the attacker can prevent the route from being established by applying one of the following modifications:

- Change the message type;
- Replace the destination IP address with another IP address;
- Decrease the destination sequence number to a smaller number;

Table 4
Possible Modifications of Fields in A RREP Message.

RREP Message Field	Modifications
Type	Change the message type.
Flags	Reverse the setting.
Prefix Size	Increase/Decrease the size of the subnet prefix.
Hop Count	Decrease it to update other nodes' forward route tables, or increase it to suppress its update.
Destination IP Address	Replace it with another IP address.
Destination Sequence Number	Increase it to update other nodes' forward route tables, or decrease it to suppress its update.
Originator IP Address	Replace it with another IP address.
Lifetime	Decrease/increase it to shorten/extend the lifetime of the route entry updated by this RREP message.

- Replace the originator IP address with another IP address;
- Decrease the lifetime field to 0;
- Replace the source IP address (in the IP header) with a non-existent IP address.

Because of the modifications of the RREP message, the originating node will receive an invalid RREP message or no RREP message at all. As a result, the originating node cannot establish a route to the destination node in this round of route discovery. However, note that the victim node may receive RREP messages from other nodes which already have routes to the destination node. In such scenarios, the attacker needs to increase the destination sequence number in the faked RREP message to suppress other valid RREP messages.

Atomic Misuse `RREP_MF_RI`

If the RREP message is the only one responding to a RREQ message, an inside attacker does not have to do anything to invade the route when the RREP message passes through it, since it is already in the route. However, if there are other RREP messages provided by other nodes, to guarantee that the RREP message through the attacker suppresses other RREP messages, the attacker may increase the destination sequence number of the RREP message by a small number. The originating node will update its route table by the faked RREP message that has the greatest destination sequence number, and thus

choose the route involving the attacker.

An inside attacker cannot isolate a node by manipulating only one RREP message. If the attacker is the only neighbor of a victim node, it can partially isolate the victim node by manipulating all the RREP messages sent to or from the victim node, which is essentially a compound misuse. Moreover, `RREP_MF` consumes little resource of the other nodes.

4.2.3 Atomic Misuse `RREP_FR`

`RREP_FR` refers to an atomic misuse with which an attacker forges a RREP message in response to a RREQ message. An inside attacker may use this misuse to disrupt a route between a victim node and a given destination, or invade a route by suppressing other alternative routes.

Atomic Misuse `RREP_FR_RD`

After receiving a RREQ message, an inside attacker may forge a RREP message as if it had a fresh enough route to the destination node. In order to suppress other legitimate RREP messages that the originating node may receive from the other nodes, the attacker may forge a faked RREP message in the following way:

- Set the destination IP address to the destination node's IP address;
- Set the originator IP address to the originating node's IP address;
- Set the source IP address (in the IP header) to a non-existent IP address;
- Set the destination IP address (in the IP header) to the node from which the attacker receives the RREQ message;
- Increase the destination sequence number by at least one, and/or decrease the hop count to 1.

The attacker unicasts the faked RREP message to the originating node along the reverse route which was established by the RREQ message. After receiving the faked RREP message, the neighbor of the attacking node will update the next hop to the destination node to the non-existent IP address in the IP header. Before the faked RREP message reaches the originating node, the originating node may have already received other legitimate RREP messages. Even in this case, the originating node will update its next hop to the destination node as the neighbor from which it receives the faked RREP message, since the faked RREP has a greater destination sequence number and/or a smaller hop count than that in the originator's route table. As a result, the later data packets from the originator to the destination node will be lost, since they will eventually be sent to a non-existent node.

Atomic Misuse `RREP_FR_RI`

If an inside attacker already has a route to the destination node, it can invade the route by unicasting a faked RREP message to the source node along the reverse route. The purpose of the attacker to still forge a RREP message is to suppress other RREP messages, possibly with shorter path from the destination node to the originating node. In order to suppress other RREP messages, the attacker can increase the destination sequence number by a small number, or decreases the hop count to 1. After receiving all the RREP messages, the originating node will update the destination sequence number in its route table to the one in the faked RREP message. It will also update the next hop to the destination node to the neighbor from which it receives the faked RREP message. As a result, the attacker can successfully be a part of the route from the originating node to the destination node.

An inside attacker cannot isolate a node by a single `RREP_FR` misuse, neither can it consume noticeable resource of the network and other nodes. Note that the impact of `RREP_FR_RC` is smaller than that caused by forging RREQ messages; RREQ messages are broadcasted throughout the network, while RREP messages are unicasted through a reverse route.

4.2.4 Atomic Misuses `RREP_AF`

In the AODV protocol, a normal node trusts all other nodes, and updates its routing table according to the received RREP messages, even if it has not generated or forwarded a corresponding RREQ message before. This gives an inside attacker further opportunities to misuse the AODV protocol.

Atomic Misuse `RREP_AF_RD`

An inside attack may disrupt an existing route in a similar way to `RREP_MF_RD` and `RREP_FR_DR`. Specifically, the attacker may create a RREP message by

- Set the type field to RREP (2);
- Set the hop count field to 1;
- Set the originator IP address as the originating node of the route and the destination IP address as the destination node of the route;
- Increase the destination sequence number by at least one;
- Set the source IP address (in the IP header) to a non-existent IP address.

Suppose the attacker already has a route to the originating node, it can then unicast the faked RREP message to the originating node. When the originating node receives the faked RREP message, it will update its route to the destination node through the non-existent node for the same reason as described in `RREP_FR_DR`.

Atomic Misuse `RREP_AF_RI`

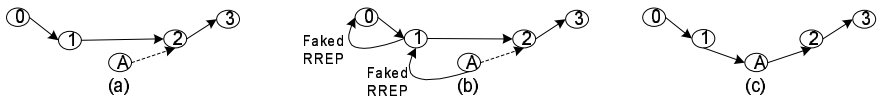


Fig. 1. An Attacker Invades a Route by Sending a Faked RREP Actively.

Fig. 2. The attacker forms a loop between node 1 and node 2 by a faked RREP message. (Node 0: originating node; nodes 1 and 2: intermediate nodes; node 3:destination node; node A: attacking node.)

It is a little surprising that an atomic misuse `RREP_AF_RI` is possible. If an inside attacker has routes to both the originating and the destination nodes of an existing route (as shown in Figure 1(a)), it can invade the route by sending a fake RREP message to the originating node. In Figure 1, assume node A is the attacking node, which already has a route to nodes 0 and 3, respectively. Node A can forge a RREP message as follows:

- Set the originator IP address to the originating node (node 0);
- Set the destination IP address to destination node (node 3);
- Set the destination sequence number to destination node (node 3)'s sequence number plus at least one;
- Set the source IP address (in the IP header) to the attacking node (node A);
- Set the destination IP address (in the IP header) to one intermediate node (node 1).

Node A then sends the faked RREP message to node 1, which forwards the faked RREP message to node 0 (Figure 1(b)). When nodes 0 and 1 receive the faked RREP message, they will update the sequence number of node 3 in their routing tables to the destination sequence number in the faked RREP message. Node 0 will still use node 1 as the next hop to node 3, but node 1 will update node A as the next hop to node 3. Note that node A already has a route to node 3. As a result, node A successfully becomes a part of the route from node 0 to node 3 (Figure 1(c)).

Atomic Misuse `RREP_AF_RC`

An inside attacker can form a loop in the network to consume resources of the nodes in the loop. As Figure 2 shows, there are two intermediate nodes, node 1 and node 2, in a route from node 0 to node 3. The attacker can form a data packets loop between node 1 and node 2 by pretending to be node 1 to

forward a RREP message from the destination node 3 to the originating node 0. The attacker may generate the faked RREP message as follows:

- Set the destination IP address to the destination node (node 3);
- Set the destination sequence number as the destination node (node 3)'s sequence number plus at least one;
- Set the originator IP address to the originating node (node 0);
- Set the source IP address (in the IP header) to one intermediate node (node 1);
- Set the destination IP address (in the IP header) to another intermediate node (node 2).

When node 2 receives the faked RREP message, it updates the next hop to node 3 as node 1. Since there is still an entry in node 2's route table that indicates the next hop to node 0 is node 1, node 2 will forward the faked RREP message to node 1, which will then forward the faked RREP message to node 0. After updating the destination sequence number in the route table, if node 0 continues to send data packets to node 3, these packets will be first sent to node 1, then node 2, and finally back to node 1 again. As a result, a loop is formed between node 1 and node 2. These data packets will be dropped until the TTL fields in the IP packets decrease to 0.

An inside attacker cannot isolate a victim node by sending out only one faked RREP message.

4.3 Atomic Misuses of RERR Messages

One RERR message may contain several unreachable destination nodes. By broadcasting one faked RERR message, an inside attacker may invalidate multiple routes, whether these routes involve the attacking node or not. It is difficult to prevent an inside attacker from misusing RERR messages. General authentication techniques will not help, since an insider has access to the authentication keys.

Table 5 summarizes the three types of atomic misuses of RERR messages and the misuse goals that they can achieve. The misuse action *Forge Reply* is not applicable to RERR messages, since RERR messages are not used to reply to any other routing messages.

4.3.1 Atomic Misuse `RERR_DR`

In order to know which neighbors should receive a RERR message, each node keeps a "precursor list" of its neighbors for each route entry. When a link

Table 5
Atomic Misuses of A RERR Message and Achievable Misuse Goals.

Atomic Misuse	Route Disruption	Route Invasion	Node Isolation	Resource consumption
RERR_DR	Yes	No	No	No
RERR_MF	Yes	No	No	No
RERR_AF	Yes	No	No	No

break is detected, the node sends a RERR message to all the nodes in the corresponding precursor list. Atomic misuses RERR_DR refer to the misuses with which an attacker simply drops a RERR message it receives without notifying its neighbors in the precursor list.

RERR_DR has limited impact on the network except for causing delays in the identification of route errors, since the upstream nodes will eventually discover the problematic routes and establish new routes.

In AODV, after receiving a RERR message, a node compares the information in the RERR message with its routing table, and it accepts the RERR message and disables a related route entry in its routing table only if the following three conditions are all satisfied:

(1) It matches unreachable destination IP address in the RERR message with a route entry's destination IP address;
(2) The next hop of the route entry equals to the source IP address (in the IP header) of the RERR message;
(3) Unreachable destination sequence number in the RERR message is greater than the destination sequence number recorded in the route entry.

Suppose an attacking node is the only neighbor in the precursor list of a node that sends a RERR message. If the attacker drops the RERR message, the upstream nodes in the precursor list of the attacker cannot receive the RERR message, and they will not be able to notify their upstream nodes about the broken link. These upstream nodes will continue to send data packets through the broken route, and the data packets will be dropped due to the broken link. However, the failures to deliver the data packets will eventually trigger the failure detection in the upstream nodes, which will find alternative routes (if they exist). Thus, this misuse has limited impact on the network.

It is easy to see that an attacker cannot invade a route, isolate a node, or consume noticeable network resource by dropping a RERR message.

4.3.2 Atomic Misuse RERR_MF

To launch RERR_MF misuses, an inside attacker may modify the RERR message after it receives a RERR message, and send the faked RERR message to the neighbors in the precursor list. The attacker may also forward modified RERR messages to those that are not in its precursor list for attack purposes. Table 6 lists the fields in a RERR message that the attacker may manipulate.

Table 6
Possible Modifications of Fields in A RERR message.

RERR Message Field	Modifications
Type	Change the value of Type.
DestCount	Modify it according to the number of unreachable destinations included in the RERR message.
Unreachable Destination IP Address	Replace it with another IP address.
Unreachable Destination Sequence Number	Increase it to update other nodes' routing table, or decrease it to suppress this entry.
Additional Unreachable Destination IP address (if needed)	Add a new destination IP address which is still reachable, or delete an unreachable IP address.
Additional Unreachable Destination Sequence number (if needed)	Increase it to update other nodes' routing table, or decrease it to suppress this entry.

Atomic Misuse RERR_MF_RD

By receiving and modifying a RERR message, an inside attacker can disrupt several routes that involve the attacker. In a faked RERR message, the attacker may replace an unreachable destination IP address with another IP address, or append new unreachable Destination IP addresses that, in fact, can be reached through the attacker. The attacker needs to increment the unreachable destination sequence number by at least one, and then broadcasts the faked RERR message to all its neighbors. If a neighbor has a route to an unreachable destination node in the faked RERR message and the next hop equals to the attacker (indicated by the source IP address in the IP header), it disables this route and updates the destination sequence number with the unreachable destination sequence number in the faked RERR message. The neighbors will then forward the faked RERR message to its neighbors in their precursor lists. As a result, all the nodes that have a route through the attacker to the destination node will disable the route. Atomic misuses RERR_MF need to be triggered by the receipt of a RERR message before modifying and forwarding

it to other nodes.

An attacker can send out a faked RERR message without being triggered by the receipt of any RERR message. Such atomic misuses are essentially `RERR_AF` misuses; we will discuss them later.

Although one faked RERR message may cause several RREQ messages broadcasted in the whole network, since atomic misuses `RERR_MF` need to be triggered by the receipt of a RERR message, it cannot consume noticeable resources of the network. An inside attacker cannot invade a route using `RERR_MF` (though in certain situations a disruption of an existing route may let other nodes choose the attacking node as a forwarding node). An inside attacker cannot isolate a victim node from sending or receiving data packets from other nodes by applying `RERR_MF`, either. If an inside attacking node is a neighbor of a victim node, it can disable all the route entries in the victim node's routing table by forwarding one faked RERR message. However, the victim node can still broadcast RREQ messages to re-establish the routes again, if there are other neighbor nodes.

4.3.3 Atomic Misuse `RERR_AF`

With `RERR_AF` atomic misuse, an inside attacker may send fake RERR messages without being triggered by any RERR message. This indeed frees an attacking node from the restriction that the attacking node must be in an existing route. As a result, an inside attacker can generate much more impact on the network with `RERR_AF` misuses than `RERR_MF`.

Atomic Misuse `RERR_AF_RD`

It is easy to see that an inside attacker may disrupt a route by sending out one faked RERR message. If the attacker is in the transmission range of an intermediate node of a route, the attacker may impersonate the intermediate node to broadcast a faked RERR message. The attacker may forge such a RERR message in the following way:

- Set the route's destination node as the unreachable destination IP address;
- Set the intermediate node's IP address as the source IP address (in the IP header);
- Set the unreachable destination sequence number as a number greater than the destination node's sequence number.

The attacker then broadcasts the faked RERR message to its neighbors. If a neighbor has a route to the destination node with the impersonated node as the next hop, it will disable the corresponding route entry. In addition, it will forward the RERR message to its upstream neighbors in its precursor list. As

a result, the routes through the intermediate node to the destination node will be disrupted.

Although the attacking node may disable several routes by using the atomic misuse `RERR_AF`, it cannot consume a noticeable amount of resources to establish new routes due to the local repair mechanism of the AODV protocol. Moreover, it is easy to see that an inside attacker cannot invade a route (`RERR_AF_RI`) or isolate a node (`RERR_AF_NI`) by sending a faked RERR message.

4.4 Atomic Misuses of RREP-ACK Messages

The Route Reply Acknowledgment (RREP-ACK) message is mainly used to prevent the uni-directional links from breaking a route. It is sent in response to a RREP message with the flag "A" bit set. Although the RREP-ACK message is quite simple, an inside attacker still has chances to misuse a RREP-ACK message to disrupt a route. Suppose there is a uni-directional link from node A to node B. When node B forwards a RREP message with "A" flag to node A, node A cannot receive the RREP message due to the uni-directional link, and thus will not send a RREP-ACK message back to node B. In normal situations, node B will realize that the link is broken. However, when an attacker overhears the RREP message from node B, it may impersonate node A to send a RREP-ACK message to node B. As a result, node B will fail to detect the uni-directional link between itself and node A.

Atomic misuses of RREP-ACK messages have very limited impact on the normal operation of the network. An insider cannot achieve any other goal that we have identified by misusing such messages.

5 Compound Misuses

In this section, we discuss the analysis of compound misuses. Unlike an atomic misuse, which can achieve a certain misuse goal through an atomic misuse of a single AODV message, a compound misuse is composed of multiple atomic misuses and possibly normal uses of the routing protocol. A component misuse of a compound misuse may be an atomic misuse that can achieve a certain goal, or simply an atomic misuse action that cannot achieve any goal if not used along with the other component actions.

We first show a compound misuse, which is aimed at invading a route between two communicating nodes through two `RREQ_AF` atomic misuses. Consider the

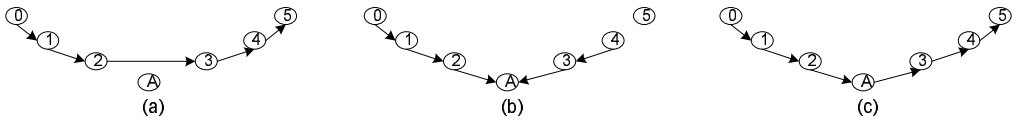


Fig. 3. Route Invasion by Two Faked RREQ Messages.

scenario shown in Figure 3(a). Suppose nodes 0 through 5 are normal nodes, and node A is a malicious node. Further assume there is a route from node 0 to node 5. In order to invade the route, node A forges the first RREQ message as follows:

- Set the originator IP address as node 5;
- Set the destination IP address as node 0;
- Set the originator sequence number to a number greater than node 5's current sequence number;
- Set the source IP address (in the IP header) as node A.

Node A then broadcasts the faked RREQ message. After receiving this message, nodes 2 and 3 will both set node A as the next hop to node 5, as in Figure 3(b). To further establish the route from node A to node 5, the attacker generates the second RREQ message as follows:

- Set the originator IP address as node A;
- Set the destination IP address as node 5;
- Set the destination sequence number to a number greater than node 5's current sequence number;
- Set the source IP address (in the IP header) as node A.

Node A then broadcasts this RREQ message, which helps node A establish a route to node 5, as shown in Figure 3(c).

It is generally difficult to manually find all possible compound misuses due to the large number of combinations of misuse or normal actions. It is certainly possible to analyze such compound misuses through automated approaches such as model checking tools. Such approaches require careful modeling of all possible atomic actions, their preconditions, and their impacts on the network. We do not do so in this paper, but consider it as possible future work. Instead, we focus on the systematic analysis of a special class of compound misuses that are launched by repeating the same type of atomic misuse actions. For convenience, we call such compound misuses *homogeneous compound misuses*. As we will see, homogeneous compound misuses may achieve more misuse goals than the corresponding atomic misuses.

An inside attacker has enough incentives to launch homogeneous compound misuses by repeating an atomic misuse. First, the routing tables of mobile nodes may change from time to time. Thus, an attacker may have to repeat an atomic misuse periodically to sustain already achieved goals, such as route disruption, route invasion, and node isolation. Secondly, most ad-hoc routing protocols such as AODV have built-in local repair mechanism, which is intended to allow mobile nodes to recover from failures. This implies that the atomic misuses targeted at disrupting services can only generate temporary impact (except for route invasion misuses, which do not trigger any failure). This recovery mechanism also forces an attacker to repeat atomic misuses to sustain the disruption of services.

Notation-wise, we extend the naming scheme for atomic misuses to denote homogeneous compound misuses. Specifically, we put an "s" after the type of routing message that is being misused in the corresponding atomic misuse. For example, `RREQs_AF` represents that an attacker actively forges multiple RREQ messages.

In our analysis, we follow the same scheme for atomic misuses to identify if a homogeneous compound misuse can achieve a misuse goal. Table 7 summarizes the analysis results. All the compound misuses at least make the corresponding misuse goal more persistent. In addition, some homogeneous compound misuses, which are underlined in Table 7, can achieve misuse goals that cannot be achieved by the corresponding atomic misuses.

Let us take a closer look at the underlined compound misuses `RREQs_MF_RC` and `RREQs_AF_RC`. As discussed earlier, the atomic misuses `RREQ_MF_RC` and `RREQ_AF_RC` cannot generate significant impact of the rest of the network. However, when an attacker broadcasts a large number of forged `RREQ` messages continuously, such messages will be rebroadcasted by the attacking node's neighbors and propagated to the rest of the network. Thus, the attacker can effectively consume the network bandwidth, power energy, and the processing time of the valid nodes.

An inside attacker can isolate a victim node from sending out data packets by applying `RREPs_FR_NI`. Whenever the attacker receives a RREQ message originated from the victim node, it sends a faked RREP message to the victim node which will send data packets through the attacker. When the attacker receives these data packets from the victim node, it just drops them. As a result, the victim node cannot send data packets to the other nodes successfully. Note that this attack does not prevent the victim node from sending packets to other nodes if there are other valid nodes that the victim node can reach.

Table 7
Homogeneous Compound Misuses and Achievable Misuse Goals

Compound Misuse	Route Disruption	Route Invasion	Node Isolation	Resource Consumption
RREQs_DR	Yes	No	No	No
RREQs_MF	Yes	Yes	Partial	Yes
RREQs_AF	Yes	Yes	Partial	Yes
RREPs_DR	Yes	No	No	No
RREPs_MF	Yes	Yes	No	No
RREPs_FR	Yes	Yes	Partial	No
RREPs_AF	Yes	Yes	Partial	Yes
RERRs_DR	Yes	No	No	No
RERRs_MF	Yes	No	No	No
RERRs_AF	Yes	No	No	Yes

An attacker can also partially isolate a node through the homogeneous compound misuse RREPs_AF_NI. Assume an inside attacker is in the transmission range of a victim node, the attacker can prevent the other nodes from receiving data packets from the victim node by sending faked RREP messages to the victim node. To achieve this goal, the attacker impersonates other nodes to send faked RREP messages to the victim node. The RREP messages are forged as follows:

- Set the destination IP address to one of the other nodes' IP address;
- Set the originator IP address to the victim node's IP address;
- Increase the destination sequence number by at least one;
- Set the source IP address (in the IP header) to the attacker's IP address.

After receiving date packets originated from the victim node, the attacker simply drops these data packets. Here, we do not set the source IP address in the IP header to a non-existent IP address, because in such case, the upstream node that forwards data packets to the non-existent node will generate a new RREQ message or a RERR message. Thus, the attacker also needs to disrupt the route from the upstream node to the destination node.

It is rather difficult to prevent the victim node from receiving data packets from other nodes by only sending faked RREP messages, because a faked RREP message is unicasted to a neighbor of the attacker, and this neighbor node may not have a route to the destination node.

Homogeneous compound misuse `RERRs_AF_RC` can effectively consume the network bandwidth and the processing time of the valid nodes. The increased routing overhead is mainly due to the messages triggered by the actively forged RERR messages. First, when a node receives a faked RERR message, it may broadcast/unicast the RERR message to other neighbors. Second, if the node wants to send data packets but the route has been invalidated by the faked RERR messages, it will have to broadcast a RREQ message. A victim node may be forced to broadcast many RREQ messages if the attacking node continuously forges RERR messages.

5.2 More Complex Compound Misuses

With atomic and the aforementioned compound misuses as building blocks, an attacker may launch more complex misuses by carefully arranging the misuses and/or normal uses of routing messages. Manually analyzing all such compound misuses is quite challenging due to the large number of combinations of atomic misuses. Fortunately, given known atomic misuses and attack goals, it is possible to automatically derive combinations of atomic misuses that may achieve these attack goals using vulnerability analysis techniques (e.g., attack graphs [12–14]). However, we consider such work outside of the scope of this paper. Our goal is to provide a set of atomic or compound misuses against the AODV protocol that can be used by the vulnerability analysis techniques as building blocks.

6 Experimental Results

In order to validate our analysis results, we have implemented all the atomic and homogeneous compound misuses and performed a series of experiments through simulation. The simulation is based on ns2 [11] with the Rice Monarch extension for the AODV protocol [17]. To take advantage of the existing AODV code, we implemented the atomic misuses by developing new agents which simply override the AODV agent's receive and send functions. Compound misuses are performed by repeating/combining the atomic misuses. This implementation, which is potentially useful for MANET intrusion detection research, can be downloaded at `http://discovery.csc.ncsu.edu/software/MisuseAODV/`.

Table 8 shows the parameters used in our experiments. We use continuous bit rate (CBR) in all our experiments. In each simulation scenario, there are 5 mobile nodes if it is for atomic misuses, and 20 nodes if it is for compound misuses. In all the experiments, there is only one inside attacker in the ad-hoc

network. The field configuration is 1000 m × 1000 m. The simulation runs for 100 simulated seconds. After arriving at a location, a node stays there for 2.0 seconds before moving to the next location. A originating node sends 4 data packets per simulated seconds. There are at most 20 connections during each simulation run. The nodes' mobility rate is 0,1,2,5, and 10 m/s. In a node's transmission range (250m), other nodes can receive signals from this node directly. The physical link bandwidth is 2 Mbps.

Table 8
Simulation Parameters

Communication Type	CBR
Number of Nodes	5 (atomic misuses) or 20 (compound misuses)
Simulation Area	1000m*1000m
Simulation Time	100 seconds
Pause Time	2.0 seconds
Packet Rate	4 pkt/sec
Number of Connections	20
Transmission Range	250m
Physical Link Bandwidth	2Mbps
Number of Inside Attackers	1

6.1 Atomic Misuses

We have verified all the atomic misuses through analyzing the trace files generated by the simulations. As an example, the fragment of the `RREP_AF_RI` trace file is as Figure 4 follows. It clearly shows that the malicious node 2 can invade the route from node 0 to node 1 after sending a faked RREP message actively.

We found that all the atomic misuses intended for *Route Disruption* and *Node Isolation* succeeded; however, the effect can last for only a short period of time. This is due to two reasons. First, the impact caused by such atomic misuses are detectable by the normal nodes, which then attempt to recover from the failures by establishing new routes. Second, all the atomic misuses are performed with a single routing message. They do not have further impact once the affected nodes perform local repair successfully.

In contrast, the atomic misuses intended for *Route Invasion* are much more subtle. Unless the routes established via atomic misuses are disrupted, the

```
...
s 7.693402932 _0_ RTR — 0 cbr 68 [0 0 0 0] ——- [0:0 1:0 30 3] [0] 0 0
r 7.695563792 _3_ RTR — 0 cbr 68 [13a 3 0 800] ——- [0:0 1:0 30 3] [0] 1 0
f 7.695563792 _3_ RTR — 0 cbr 68 [13a 3 0 800] ——- [0:0 1:0 29 4] [0] 1 0
r 7.697885792 _4_ RTR — 0 cbr 68 [13a 4 3 800] ——- [0:0 1:0 29 4] [0] 2 0
f 7.697885792 _4_ RTR — 0 cbr 68 [13a 4 3 800] ——- [0:0 1:0 28 1] [0] 2 0
r 7.700206910 _1_ AGT — 0 cbr 68 [13a 1 4 800] ——- [0:0 1:0 28 1] [0] 3 0
...
s 10.671253994 _2_ RTR — 0 AODV 44 [0 0 0 0] ——- [2:255 0:255 30 3] [0x4 1 [1 7] 10.000000] (REPLY)
r 10.680841103 _3_ RTR — 0 AODV 44 [13a 3 2 800] ——- [2:255 0:255 30 3] [0x4 1 [1 7] 10.000000] (REPLY)
f 10.680841103 _3_ RTR — 0 AODV 44 [13a 3 2 800] ——- [2:255 0:255 29 0] [0x4 2 [1 7] 10.000000] (REPLY)
r 10.682829964 _0_ RTR — 0 AODV 44 [13a 0 3 800] ——- [2:255 0:255 29 0] [0x4 2 [1 7] 10.000000] (REPLY)
...
s 11.619037018 _0_ AGT — 17 cbr 48 [0 0 0 0] ——- [0:0 1:0 32 0] [15] 0 0
r 11.619037018 _0_ RTR — 17 cbr 48 [0 0 0 0] ——- [0:0 1:0 32 0] [15] 0 0
s 11.619037018 _0_ RTR — 17 cbr 68 [0 0 0 0] ——- [0:0 1:0 30 3] [15] 0 0
r 11.620773878 _3_ RTR — 17 cbr 68 [13a 3 0 800] ——- [0:0 1:0 30 3] [15] 1 0
f 11.620773878 _3_ RTR — 17 cbr 68 [13a 3 0 800] ——- [0:0 1:0 29 2] [15] 1 0
r 11.623295681 _2_ RTR — 17 cbr 68 [13a 2 3 800] ——- [0:0 1:0 29 2] [15] 2 0
f 11.623295681 _2_ RTR — 17 cbr 68 [13a 2 3 800] ——- [0:0 1:0 28 4] [15] 2 0
r 11.625537484 _4_ RTR — 17 cbr 68 [13a 4 2 800] ——- [0:0 1:0 28 4] [15] 3 0
f 11.625537484 _4_ RTR — 17 cbr 68 [13a 4 2 800] ——- [0:0 1:0 27 1] [15] 3 0
r 11.627638602 _1_ AGT — 17 cbr 68 [13a 1 4 800] ——- [0:0 1:0 27 1] [15] 4 0
```

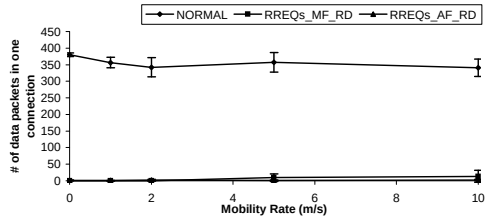
Fig. 4. Fragments of the trace file of `RREP_AF_RI` misuse

victim nodes will continue to use the routes involving the inside attacker to transmit data packets. Details of the experiments about atomic misuses can be found in the related technical report [18].

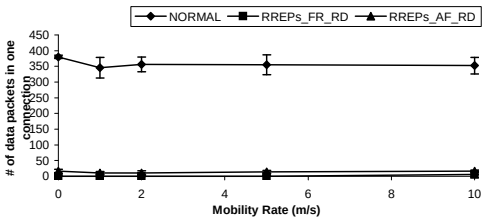
6.2 Homogeneous Compound Misuses

Though atomic misuses for route disruption and node isolation do not last very long when they are used individually, our experiments show that they are quite powerful when they are used repeatedly as homogeneous compound misuses.

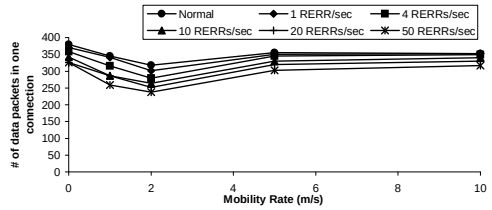
Let us first look at the simulation results for `RREQs_DR_RD`, `RREPs_DR_RD` and `RREPs_MF_RD`. If an attacking node is the only node between two parts of an ad-hoc network, it may separate the nodes in these two parts by using the aforementioned compound misuses. From our simulation results, we discover that if the attacking node only drops the RREQ messages from the originating node to the destination node (through `RREQs_DR_RD`), the originating node still has chances to establish a route to the destination node. The reason is that the nodes in the same side of the originating node may establish a route to the destination node by broadcasting RREQ messages to the destination node or receiving RREQ messages originated from the destination node. These nodes can send RREP messages to the originating node. Therefore, in `RREQs_DR_RD` misuse, the attacking node needs to drop all the RREQ messages whose Originator IP Address or Destination IP Address is set to the destination node. In `RREPs_DR_RD`, the route cannot be established in most scenarios except when



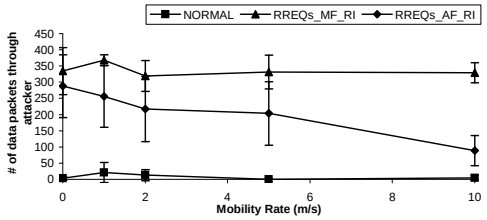
(a) Route Disruption by RREQs



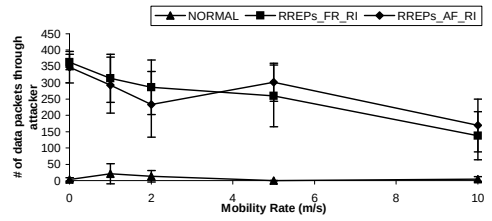
(b) Route Disruption by RREPs



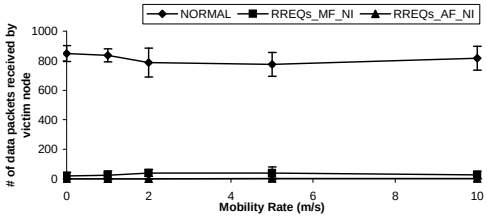
(c) Route Disruption by RERRs



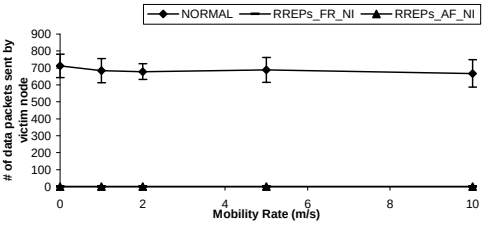
(d) Route Invasion by RREQs



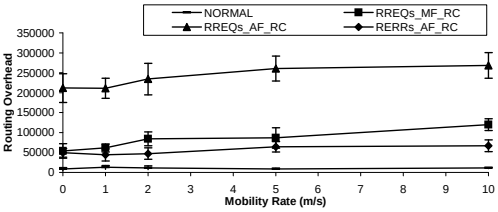
(e) Route Invasion by RREPs



(f) Node Isolation by RREQs



(g) Node Isolation by RREPs



(h) Resource Consumption by RREQs and RERRs

Fig. 5. Experimental Results about Homogeneous Compound Misuses

the originating node wants to send data packets to the destination node right after adding a route entry (due to the receipt of a RREQ message from the destination node). The compound misuse `RREPs_MF_DR` can prevent this by increasing the destination sequence number in the faked RREP messages.

Figure 5 shows additional experimental results for homogeneous compound

misuses. Each data point in these figures is an average of 10 simulation runs with identical configuration but different randomly generated simulation scenarios. The Y axis error bars show confidence interval at 95% confidence.

Figure 5(a) displays the numbers of data packets transmitted between two victim nodes when there is no attack and when an attacking node uses compound misuses RREQs_MF_RD and RREQs_AF_RD. It shows that when RREQs_MF_RD and RREQs_AF_RD are used against these two nodes, the number of data packets being transmitted between them drops almost to zero. Thus, RREQs_MF_RD and RREQs_AF_RD can successfully disrupt routes. Figure 5(b) shows the same effect when compound misuses RREPs_FR_RD and RREPs_AF_RD are used.

From an inside attacker's perspective, there are some tradeoffs when choosing different homogeneous compound misuses to disrupt a route. RREQs_MF_RD and RREPs_FR_RD require the attacker receive corresponding RREQ messages prior to launching the attacks. RREQs_AF_RD is much easier to apply than the above compound misuses, but it is prone to detection due to the actively forged RREQ messages. RREPs_AF_RD needs the attacker to have a route to the victim node before sending the faked RREP message.

By forging RERR messages, an attacker may disrupt a route; however, due to the local repair and the data packets buffering of the intermediate nodes, if the faked RERR messages' rate is not high enough, the data packets can still be transmitted to the destination node with a little delay, due to the data buffer at each node. In our simulation, the data buffer size at each node is 64 packets, and the maximum lifetime of each packet in the buffer is 30 seconds. Figure 5(c) shows how the RERR message rate affects the throughput of the data packets in RERRs_AF_DR. When the attacker only sends 1 faked RERR message per second, it can hardly impair the transmission of data packets. The throughput decreases when the attacker sends out faked RERR messages at higher rates. However, 80% data packets can still reach the destination node even if the attacker sends 50 faked RERR messages per second. This shows that the local repair mechanism of the AODV protocol can handle failures due to natural faults and misuses of RERR messages fairly well.

Figures 5(d) and 5(e) show the experimental results about several homogeneous compound misuses intended for route invasion. Figure 5(d) shows the number of data packets transmitted through an inside attacker under normal situation and when RREQs_MF_RI and RREQs_AF_RI are used. In normal situations, there are nearly zero data packets transmitted through the attacking node. However, when these two compound misuses are used, the number of data packets transmitted through the attacking node is significantly increased. That is, the RREQs_MF_RI and RREQs_AF_RI effectively make the attacker a part of the route between the two victim nodes. In RREQs_AF_RI, the number of data packets through the attacker drops when the mobility rate increases.

It is mainly due to the collisions in the MAC layer introduced by broadcasting many fake RREQ messages during a small period of time.

Figure 5(e) shows the number of data packets transmitted through the attacking node under normal situations and when RREPs_FR_RI and RREPs_AF_RI are used. It shows similar results to Figure 5(d). When these misuses are used, most data packets sent between two victim nodes go through the attacking node. The number of data packets drops when the mobility rate increases.

Figures 5(f) and 5(g) include the experimental results about node isolation. Figure 5(f) shows the number of data packets received by a victim node under RREQs_MF_NI and RREQs_AF_NI misuses. It is easy to see that when these compound misuses are used against a victim node, the victim node can hardly receive data packets from other nodes. Figure 5(g) shows the number of data packets sent from a victim node and finally received by the destination nodes when RREPs_FR_NI and RREPs_AF_NI are used. It shows that other nodes can hardly receive any data packets from the victim node in these cases. Indeed, by combining some of these compound misuses together, the attacker may effectively isolate a victim node.

An attacking node may isolate a victim node by using RREQs_DR_NI and RREPs_DR_NI only if the attacking node is the only neighbor of the victim node; however, due to the mobility of the ad-hoc networks, the victim node may have other neighbors after moving to a new location.

Figure 5(h) shows the experimental results about homogeneous compound misuses intended for resource consumption. We use the total number of routing messages to measure the routing overhead. Figure 5(h) shows that both RREQs_MF_RC and RREQs_AF_RC significantly increase the routing overhead in the network. In our experiment with RREQs_AF_RC, the attacking node generates 20 faked RREQ messages per second. Given the 100 seconds simulation duration, the attacking node totally generates 2,000 faked RREQ messages. As a result, the number of routing packets in the network increases to over 200,000 packets from fewer than 10,000. Though RREQs_MF_RC requires being triggered by normal RREQ messages, our experiments also show this misuse significantly increases the routing overhead. In RERRs_AF_RC, the attacking node also generates 20 faked RERR messages per second, while each RERR message includes all the 20 nodes in the network as the unreachable destination IP addresses. The simulation result shows that RERRs_AF_RC can also significantly increase the routing overhead. Our experiments also indicate that the routing overhead slightly increases with the nodes' mobility rate.

Compound misuses RREPs_AF_RC consume nodes' resource by forming a loop among them. To understand the impact of this misuse, we compare the energy consumption of the nodes involved in the loop with the energy consumption

where there is no such misuse. In ns2's energy model, it consumes a node 0.3 watt to receive a packet, and 0.6 watt to send a packet. During the 100 second simulation time, when a loop appears in the network, the nodes in the loop on average consumes 8.066128 joules energy, while they only consume on average 0.8112445 joules energy if in the loop-free case. In other words, `RREPs_AF_RC` costs the nodes in the loop about 10 times more energy than the normal cases. Moreover, the data packets transmitted in the loop will be dropped in the end.

7 Related Work

Research in MANET has been rather active. Several routing protocols (e.g., AODV [9], DSR [19], DSDV [20], ZRP [21], LAR [22]) have been proposed to discover and maintain routes in MANET environments. The focus of earlier research is on the various mechanisms to maintain routes and improve the performance of the routing protocols. However, most of the early protocols do not consider the security of the routing message.

Attacks against ad-hoc routing protocols have been studied in the past. Hubaux et al. identified attacks on basic mechanisms and on security mechanisms of MANET [8]. Hu, Perrig, and Johnson [4] formalized an attack model using the number of compromised nodes and the number of compromised cryptographic keys. Karlof and Wagner [23] summarized several attacks on sensor network routing protocols, and suggested possible countermeasures for these attacks. Our work in this paper complements the above research by providing a systematic analysis of insider attacks against the AODV protocol.

Several secure routing protocols have been proposed to prevent or detect attacks against routing protocols. The earlier works attempt to solve the security problem by using public key cryptography. Sanzgiri et al [2] proposed to use digital signatures to authenticate the routing messages and a trusted certificate server to facilitate key and trust managements for mobile nodes. Zhou and Haas [24] proposed to use a key management service that distributes the certification authority into several servers by employing threshold cryptography, which can tolerate server failures and potentially colluding attackers to a certain extent. Hubaux et al developed a self-organizing, distributed public-key infrastructure [8], which provides probabilistic guarantees for mobile nodes to discover certificate paths leading to the other nodes. Zapata [1] proposed to employ public key cryptography to authenticate the AODV routing messages, and use one-way hash chain to protect the variant "hop count". It differs from Sanzgiri et al [2]'s proposal in that only the end-points of a route provide signatures on the routing messages, while the intermediate nodes verify and forward the messages without appending their signatures. A common limitation of the above approaches is that they all depend on public key cryptography.

Since a typical mobile node (e.g., PDA) is usually less powerful than desktop computers, and is often powered by batteries, these proposals are potentially too expensive in terms of computation and energy consumption. By providing authentication mechanisms on the routing messages, these proposals can prevent external attackers from impersonating other nodes. However, they are still vulnerable to attacks from inside attackers, who may compromise and/or impersonate a node.

Recent secure routing protocols usually use symmetric cryptography to authenticate the routing messages. Papadimitratos and Haas proposed to authenticate the route discovery process with a secret key shared between the source and the destination nodes [3]. Basagni et al. employed a network-wide secret key to secure the routing messages [25]. They proposed a key management scheme to periodically update the secret key used by all nodes. This approach is efficient, but it is vulnerable to a single point of compromise. Yi et al. modified AODV to include security metrics for route discovery, using different trust levels with a shared symmetric key for each level [26]. These proposals are also vulnerable to inside attackers. When an inside attacker grasps all or part of the secret keys, it can launch all the misuses listed in this paper.

Hu, Perrig, and Johnson have proposed a sequence of secure MANET routing protocols, including Ariadne [4] and SEAD [5], as well as security mechanisms for routing protocols [27]. Their techniques include authenticating routing messages through a one-way key chain with delayed disclosures of keys and authentication code with secret keys shared by mobile nodes. They introduced some general attacks on ad-hoc routing protocols in [4]. Our work extends their attack model by employing atomic misuse actions into our analysis scheme, and summarizing four prominent misuse goals. They also identified a link-layer attack called wormhole attack and proposed to use one-way key chain with tight time synchronization to discard stale packets [28].

Intrusion detection can provide another layer of protection for MANET. Zhang and Lee proposed a distributed and cooperative IDS architecture in mobile ad-hoc networks [29]. They use data on the node's physical movements and the corresponding change in its routing table as the trace data to build the anomaly detection model. In Marti et al.'s proposal [7], each node uses a component called *watchdog* to detect misbehaving nodes, and another component called *pathrater* to choose a reliable route based on the information collected by the watchdog. Buchegger and Boudec [6] extended Marti et al.'s proposal by allowing each node not only to monitor the bad behaviors of it neighbors, but to collect the list of malicious nodes from warnings sent from other trusted nodes. Our analysis result can be used to build the misuse model for intrusion detection systems.

There have been several works on network vulnerability analysis (e.g., [12–16]), which employ model checking techniques to identify sequences of component attacks that may lead to the compromise of certain security properties. Our work in this paper is focused on the component insider attacks, and thus provides building blocks that may be used by these tools. We consider these works as complementary to ours.

8 Conclusion and Future Work

In this paper, we presented a scheme to analyze insider attacks against mobile ad-hoc routing protocols, and reported a systematic analysis of the AODV protocol. We classified the possible insider attacks into atomic misuses and compound misuses, and identified a number of atomic misuses as well as compound misuses. We also performed a series of experiments (based on simulation) to validate these misuses. Our results showed that an inside attacker can effectively invade into routes, consume the nodes' resources, isolate victim nodes from the rest of the network, disrupt existing route, or prevent certain nodes from establishing routes in AODV networks. The results in this paper are potentially useful for protocol developers to evaluate their designs, and for intrusion detection researchers to validate their detection algorithms and systems.

It is not surprising to find these attacks against the AODV protocol, since this protocol was not designed with security as a goal. However, our results indicate that achieving security is not as simple as combining generic security mechanisms with a target application such as the AODV protocol. Protocol and application designers must pay special attention to the security requirements, the risks and threats, as well as the semantics of the specific protocols and applications to have truly secure and dependable solutions.

Our experience also shows providing security in MANET is a quite challenging task, particularly due to the fact that mobile nodes in MANET are subject to capture and compromise. The difficulty of MANET security also suggests that additional mechanism such as intrusion detection should be used along with prevention based security mechanisms such as authentication and encryption to accommodate possible failures of the prevention based mechanisms.

The results in this paper represent our initial attempt in understanding insider attacks against MANET routing protocols. In our future research, we plan to investigate how to use these results to facilitate intrusion detection in MANET. Moreover, to understand how well the secure MANET routing protocols can withstand these attacks, we plan to further investigate insider attacks against secure MANET routing protocols such as SAODV [1] and ARAN [2].

References

[1] M. G. Zapata, Internet Draft, draft-guerrero-manet-saodv-00.txt (October 2001).

[2] 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 Tenth IEEE International Conference on Network Protocols (ICNP), 2002.

[3] P. Papadimitratos, Z. J. Haas, Secure routing for mobile ad hoc networks, in: Proceedings of the SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CDNS 2002), 2002, pp. 27 – 31.

[4] Y. Hu, A. Perrig, D. B. Johnson, Ariadne: A secure on-demand routing protocol for ad hoc networks, in: Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking (MobiCom '02), 2002.

[5] Y. Hu, D. B. Johnson, A. Perrig, SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks, in: Fourth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '02), 2002.

[6] S. Buchegger, J. L. Boudec, Performance analysis of the CONFIDANT protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks), in: Proceedings of The Third ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2002, pp. 226–236.

[7] S. Marti, T. J. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, in: Proceedings of the Sixth annual ACM/IEEE International Conference on Mobile Computing and Networking, 2000, pp. 255–265.

[8] J. Hubaux, L. Buttyan, S. Capkun, The quest for security in mobile ad hoc networks, in: Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing, 2001.

[9] C. Perkins, E. Belding-Royer, S. Das, Ad hoc on-demand distance vector (AODV) routing, Internet Draft, draft-ietf-manet-aodv-13.txt (February 2003).

[10] S. Kent, R. Atkinson, IP authentication header, IETF RFC 2402 (November 1998).

[11] The network simulator – ns-2, `http://www.isi.edu/nsnam/ns/`.

[12] O. Sheyner, J. Haines, S. Jha, R. Lippmann, J. Wing, Automated generation and analysis of attack graphs, in: Proceedings of IEEE Symposium on Security and Privacy, 2002.

[13] P. Ammann, D. Wijesekera, S. Kaushik, Scalable, graph-based network vulnerability analysis, in: Proceedings of the 9th ACM Conference on Computer and Communications Security, 2002, pp. 217–224.

[14] S. Jha, O. Sheyner, J. Wing, Two formal analyses of attack graphs, in: Proceedings of the 15th Computer Security Foundation Workshop, 2002.

[15] C. Phillips, L. Swiler, A graph-based system for network vulnerability analysis, in: Proceedings of New Security Paradigms Workshop, 1998, pp. 71–79.

[16] L. Swiler, C. Phillips, D. Ellis, S. Chakerian, Computer-attack graph generation tool, in: Proceedings of the DARPA Information Survivability Conference andn Exposition, 2001, pp. 307–321.

[17] The rice university monarch project: Mobile networking architectures, `http://www.monarch.cs.rice.edu`.

[18] P. Ning, K. Sun, How to misuse AODV: A case study of insider attacks against mobile ad-hoc routing protocols, Tech. Rep. TR-2003-07, North Carolina State University, Department of Computer Science (February 2003).

[19] D. B. Johnson, D. A. Maltz, Y. Hu, J. G. Jetcheva, Internet Draft, draft-ietf-manet-dsr-07.txt (February 2002).

[20] C. Perkins, P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, in: ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, 1994, pp. 234–244.

[21] Z. J. Hass, M. R. Pearlman, P. Samar, Internet Draft, draft-ietf-manet-zone-zrp-04.txt (July 2002).

[22] Y. B. Ko, N. Vaidya, Location-aided routing (lar) in mobile ad hoc networks, in: Proceedings ACM/IEEE MOBICOM 98), 1998, pp. 66–75.

[23] C. Karlof, D. Wagner, Secure routing in wireless sensor networks: Attacks and countermeasures, in: First IEEE International Workshop on Sensor Network Protocols and Applications, 2003.

[24] L. Zhou, Z. J. Haas, Securing ad hoc networks, IEEE Network 13 (6) (1999) 24–30.

[25] S. Basagni, K. Herrin, D. Bruschi, E. Rosti, Secure pebblenets, in: Proceedings of ACM International Symposium on Mobile ad hoc networking and computing, 2001, pp. 156–163.

[26] S. Yi, P. Naldurg, R. Kravets, Security-aware routing protocol for wireless ad hoc networks, in: Proceedings of ACM Symposium on Mobile Ad Hoc Networking & Computing (Mobihoc 2001), 2001.

[27] Y. Hu, A. Perrig, D. V. Johnson, Efficient security mechanisms for routing protocols, in: Proceedings of the 10th Annual Network and Distributed System Security Symposium, 2003, pp. 57–73.

[28] Y. Hu, A. Perrig, D. Johnson, Packet leashes: A defense against wormhole attacks in wireless ad hoc networks, in: Proceedings of INFOCOM 2003, 2003.

[29] Y. Zhang, W. Lee, Intrusion detection in wireless ad hoc networks, in: Proceedings of the 6th International Conference on Mobile Computing and Networking (MobiCom 2000), 2000, pp. 275–283.