Security in Ad-Hoc Routing Protocols

Prof. Deshpande Vivek S.
Lecturer, Maharashtra Institute Of Technology Women Engineering.
Pune, Maharashtra, India.
E-Mail: vsd_deshpande@rediffmail.com

Abstract

Mobile Ad-Hoc Networks (MANETs) are becoming increasingly popular as more and more mobile devices find their way to the public, besides "traditional" uses such as military battlefields and disaster situations they are being used more and more in every-day situations. With this increased usage comes the need for making the networks secure as well as efficient, something that is not easily done as many of the demands of network security conflicts with the demands on mobile networks due to the nature of the mobile devices (e.g. low power consumption, low processing load). The concept and structure of MANETs make them prone to be easily attacked using several techniques often used against wired networks as well as new methods particular to MANETs. Security issues arise in many different areas including physical security, key management, routing and intrusion detection, many of which are vital to a functional MANET.

In this paper we focus on the security issues related to ad hoc routing protocols in particular. The routing in ad hoc networks remains a key issue since without properly functioning routing protocols, the network simply will not work the way it's intended to. Unfortunately, routing may also be one of the most difficult areas to protect against attacks because of the ad hoc nature of MANETs. We will present the main security risks involved in ad-hoc routing as well as the solutions to these problems that are available today.

Ad-Hoc Routing Protocols

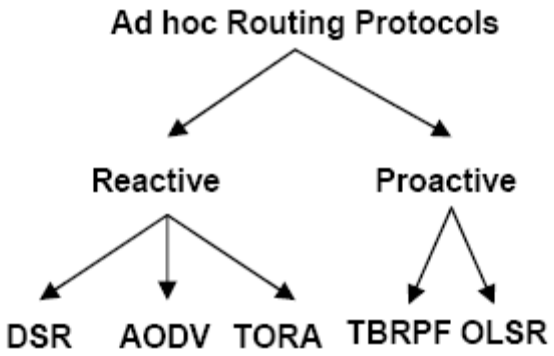
1) General considerations about ad-hoc routing protocols

a) Types of ad-hoc Routing protocols

There are mainly two types of ad-hoc routing protocols:

1. Proactive routing protocols, where the nodes keep updating their routing tables, by sending periodical messages. We have, for example, OLSR (Optimized Link State Routing protocol) and TBRPF (Topology Broadcast based on Reverse Path Forwarding)

2. Reactive (On Demand) routing protocols, where routes are created only when needed. We have, for example, DSR (Dynamic Source Routing protocol) and AODV (Ad hoc On-Demand Distance Vector Routing protocol)



Examples of adhoc routing protocols

b) Desired characteristics of ad-hoc routing protocols

- Ad-hoc routing protocols have some special requirements:
- Distributed operation
- Loop freedom
- Demand-based operation
- Sleep period operation
- Unidirectional link support
- Security

c) Ad-hoc network's security characteristics

The characteristics of the Ad-hoc network's security are,

- Availability
- Confidentiality
- Integrity
- Authentication
- Non-repudiation

2) Problems with ad-hoc routing protocols: causes

In ad-hoc routing protocols, nodes exchange information with each other about the network topology, because the nodes are also routers. This fact is also an important weakness because a compromised node could give bad information to redirect traffic or simply stop it. Moreover, we can say that routing protocols are very brittle in term of security. This part aims to provide a description of the causes of the problems with ad-hoc routing protocols.

a) Infrastructure of ad-hoc networks:

Ad-hoc networks have no predetermined fixed infrastructure, that's why the nodes themselves have to deal with the routing of packets. Each node relies on the other neighboring nodes to route packets for them.

b) Dynamic topology of ad-hoc networks:

The organization of the nodes may change because of the mobility-aspect of ad-hoc networks: they contain nodes that may frequently change their locations. Because of this fact, we talk about the dynamic topology of these

networks, which is a main characteristic that causes problems: when several ad-hoc networks mix together, there can be duplications of IP addresses, and resolving it is not so simple. Then, attacks can easily occur by using this duplication of IP address (cf. attacks using impersonation)

c) Problems associated with wireless communication:

Wireless channels have a poor protection to noise and signal interferences, therefore routing related control messages can be tampered. A malicious intruder can just spy on the line, jam, interrupt or distort the information circulating within this network.

d) Implicit trust relationship between neighbors:

Actual ad-hoc routing protocols suppose that all participants are honest. Then, this directly allows malicious nodes to operate and try to paralyze the whole network, just by providing wrong information.

Possible Attacks In Ad-Hoc Routing Protocols

Due to their particular architecture, ad-hoc networks are more easily attacked than wired network. We can distinguish two kinds of attack: the passive attacks and the active attacks. A passive attack does not disrupt the operation of the protocol, but tries to discover valuable information by listening to traffic. Instead, an active attack injects arbitrary packets and tries to disrupt the operation of the protocol in order to limit availability, gain authentication, or attract packets destined to other nodes.

The routing protocols in MANET are quite insecure because attackers can easily obtain information about network topology. Indeed in AODV and DSR protocols, the route discovery packets are carried in clear text. So a malicious node can discover the network structure just by analyzing this kind of packets and may be able to determine the role of each node in the network. With all these information more serious attacks can be performed in order to disturb the network operation by isolate important nodes, etc. Let us see the different attacks possible by using modification first, then by using impersonation and finally the attacks using fabrication.

1) Attacks using modification:

One of the simplest ways for a malicious node to disturb the good operation of an ad-hoc network is to announce better routes (to reach other nodes or just a specific one) than the other nodes. This kind of attack is based on the modification of the metric value for a route or by altering control message fields (Denial Of Service attacks).

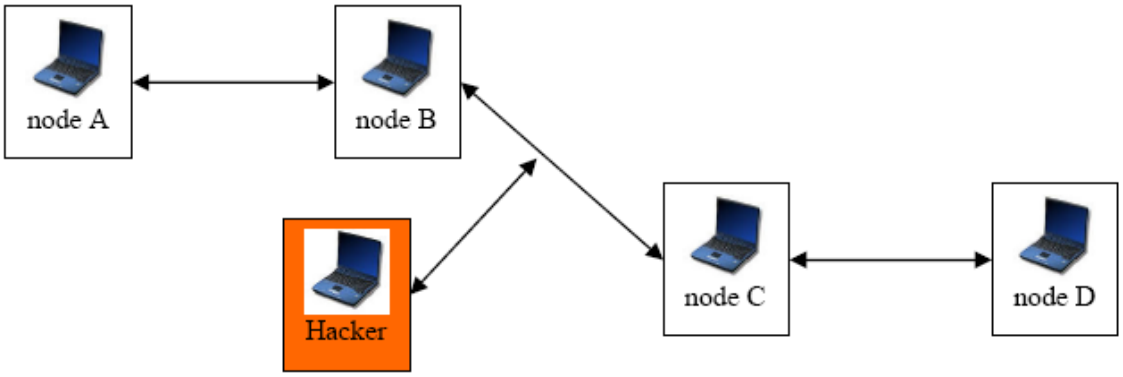


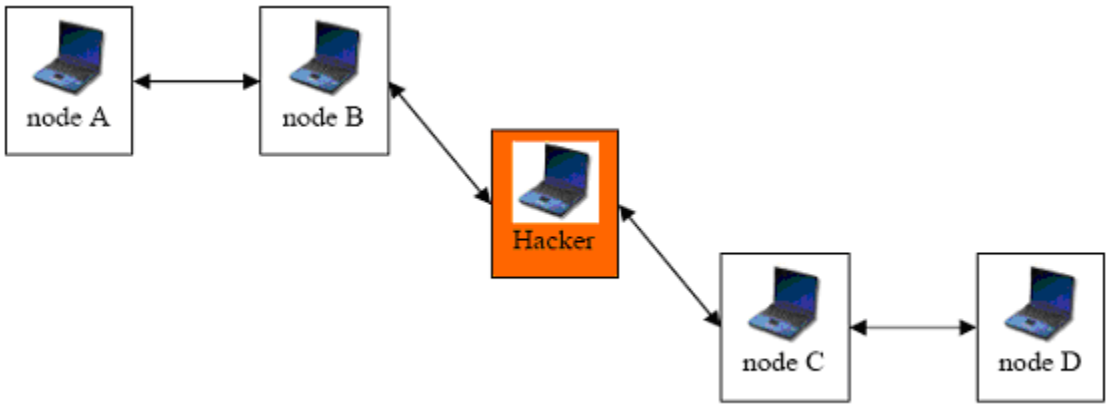
Figure 1

For example, in the network illustrated in the Figure 1, a malicious node "Hacker" could keep traffic from reaching the node D by consistently advertising to the node B a shorter route to the node D than the route to D that C is advertising.

a) Redirection by changing the route sequence number

In ad-hoc networks, like in wired networks, the better path to reach a destination node is determined by a specific value, which is the metric and is often the element, which determines the better route. Smaller this value is, better is the route. That's why a simple way to attack a network is to change this value with a smaller number than the last "better" value.

In the figure 1 we have shown that a malicious node called "Hacker" try to insert itself to the network in order to disturb its operation. When the node A wants to communicate with the node D, it broadcasts a message asking all the nodes around the better path to reach the node D. B will received the message and forward it. The node C will reply that it has a direct route to D and in this reply message; it will give a value for the metric. Now if the malicious node replies to the node B too that it has a direct route to the node D with a smaller metric value than C, B will consider this route as the best one and delete the path with the node C. The result in the example is shown in the figure 2 below.



b) Redirection with modified hop count (specific to AODV protocol)

When a node cannot decide what the best route is regarding to different metrics, it can use the number of hops to decide which path is the best route to reach a specific node. This is the case in the AODV protocol. In this case, the protocol uses the hop count value to determine the best route. Also a malicious node can disturb the network too, by announcing a smallest hop count value to reach the node. In general, hackers use the value zero to be sure to have the smallest hop count value.

c) Denial Of Service (DOS) attacks with modified source routes

The DOS attack is well-known in computer security and can be efficient in ad-hoc networks without secure routing protocols. A simple way to understand the operation of DOS attacks is to see the figure 2.

In this figure, a malicious node is located in the network. If the node A wants to communicate with the node E, it sends data packets following its route cache to the node E including the malicious node. Also when the malicious node will receive the data

packets, it can change the header of these packets in order to abort the transmission of the data.

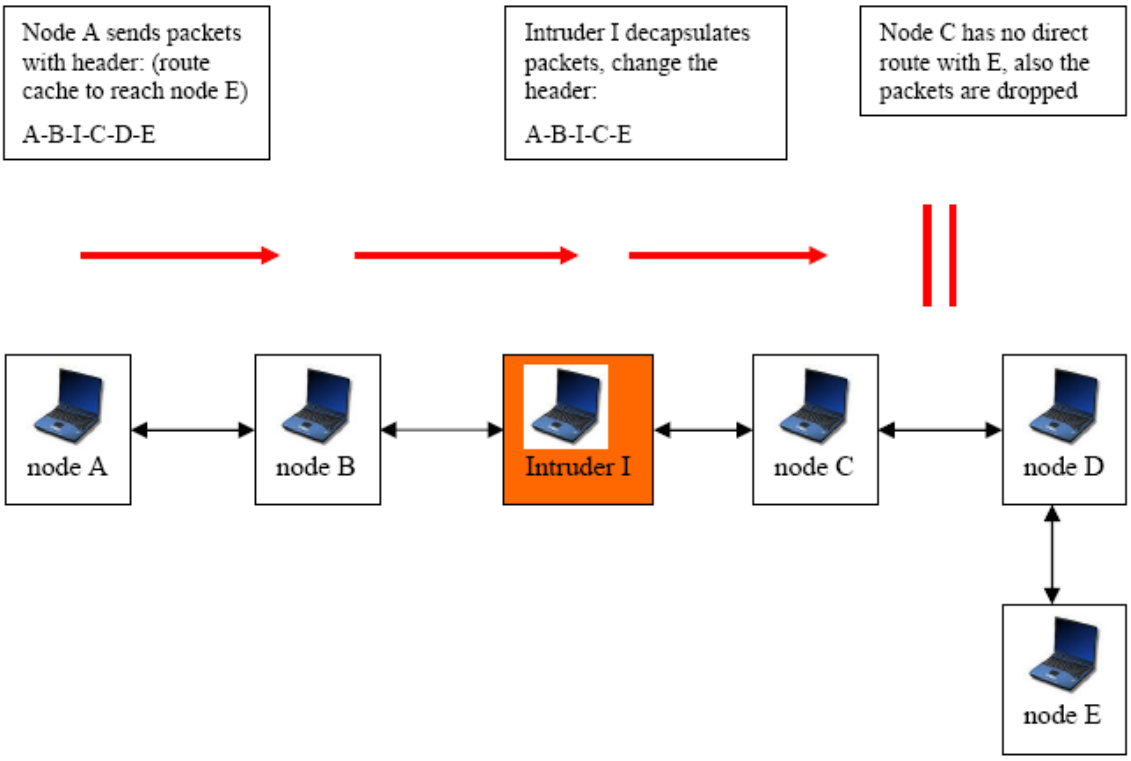


Figure 2

2) Attacks using impersonation

These attacks are called spoofing since the malicious node hide its real IP address or MAC address and uses another one. As current ad-hoc routing protocols like AODV and DSR do not authenticate source IP address, a malicious node can launch many attacks by using spoofing. For example, a hacker can create loops in the network to isolate a node from the remainder of the network. To do this, the hacker just has to take IP address of other node in the network and then use them to announce new route (with smallest metric) to the others nodes. By doing this, he can easily modify the network topology as he wants.

3) Attacks using fabrication

We can distinguish three kinds of attacks using fabrication. They are,

a) Falsifying route error messages

The first attack is quite common in AODV and DSR because these two protocols are using path maintenance to recover the good path when some nodes have moved. The weakness of this architecture is that when a node moves, the closest node sends an "error" message to the others to inform them that the route is no more available. If a malicious node usurps the identity of another node by using spoofing and send error messages to the others, the other nodes will update their routing tables with this information. Also the malicious node may insulate any node quite easily.

b) Corrupting routing state: route cache poisoning

This is a passive attack that can occur in DSR especially because of the promiscuous mode of updating routing table which is employed by DSR. This occurs

when information stored in routing table at routers is deleted, altered or injected with false information. Indeed, in addition to learning routes from headers of packets, which a node is processing along a path, routes in DSR may also be learned from promiscuously received packets. A node overhearing any packet may add the routing information contained in that packet's header to its own route cache, even if that node is not on the path from source to destination.

The vulnerability of this system is that an attacker could easily exploit this method of learning routes and poison route caches. For example, the hacker just has to broadcast a message with a spoofed IP address in the other nodes around. When they will receive this message, the nodes would add this new route to their cache and also communicate now with this route to reach a special node (the malicious node in fact instead of the one with the same IP address as the hacker's node).

c) Routing table overflow attack

If the ad-hoc network is using a "proactive" protocol, it means that the protocol algorithm try to find routing information even before they are needed. (Instead of "reactive" protocol which do this after). This is a vulnerability used by this attack, because the attacker attempts to create route to non-existent nodes. If he creates enough routes, new routes cannot be created anymore because of an overwhelming pressure of the protocol.

d) Other attacks using fabrication

Replay attack: an attacker sends old advertisements to a node causing it to update its routing table with stale routes.

Black hole: an attacker advertises a zero metric for all destinations causing all nodes around it to route packets towards it.

Solutions To Security Problems In Ad-Hoc Routing Protocols

In order to provide solutions to the security issues involved we must first establish that there are different kinds of ad-hoc networks and the different types of networks put different demands on the infrastructure and also determines what means are available to improve security. In ad-hoc networks are divided into the following categories:

• Open

This type of environment is characterized by the lack of any infrastructure that one can use in order to maintain security. The nodes present in an open environment can be of any type and not necessarily known beforehand. Therefore any kind of central authority system that requires prior knowledge of the nodes in the network is not going to work. Typically this is not a very common environment and the extreme openness it presumes also limits the available security measures a great deal.

• Managed-Open

The managed-open environment is probably the one where most research is being done today as it is the type of environment we are most likely to see expand in the nearest future .In this type of environment there the possibility to use already established infrastructure to some extent to help us secure the ad-hoc network. This opens up a whole new range of strategies using certificate servers and other similar software to provide a starting point of the security in the network.

• Managed-Hostile

This is perhaps the classic ad-hoc environment and it's described as nodes in a military war-zone, or perhaps in a disaster area. Here security is the primary goal and even information such as the location of the nodes involved is considered very sensitive information. In this type of environment security is considered to be much more important than performance and as such the security measures can be made a bit more extreme.

Depending on the type of network environment, different types of security-enhancing techniques have been developed, each of which tries to minimize the security risks while still keeping within the bounds set up by the particular environment. There are two main different approaches to designing the techniques: adding enhancements to existing protocols and creating new protocols from the ground up. They are,

1) Protocol enhancements

These techniques are basically enhancements that, if not mentioned otherwise, can be applied to any of the current ad-hoc routing protocols in use today.

a) Security-Aware ad hoc Routing, SAR

SAR is an attempt to use traditional shared symmetric key encryption in order to provide a higher level of security in ad-hoc networks. SAR can basically extend any of the current ad hoc routing protocols without any major issues.

While current ad hoc routing protocols are successful at finding the shortest path to any node within the network, SAR extends this function by finding the shortest path providing a requested trust level. The different trust levels are implemented using shared symmetric keys. In order for a node to forward or receive a packet it first has to decrypt it and therefore it needs the required key. Any nodes not on the requested trust level will not have the key and cannot forward or read the packets

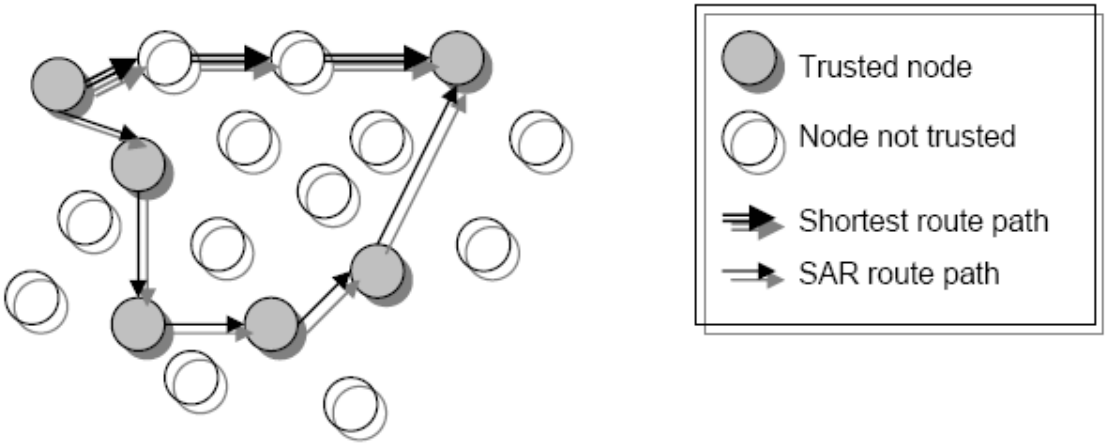


Figure 4

Figure 4 illustrates the difference in route path selection between SAR and a normal ad-hoc routing protocol choosing the shortest routing path. The arrows represent route hops.

Every node sending a packet decides what trust level to use for the transfer and thereby decides the trust level required by every node that will forward the packet to its final destination.

SAR is indeed secure in the way that it does ensure that only nodes having the required trust level will read and reroute the packets being sent. Unfortunately, SAR still leaves a lot of security issues uncovered and still open for attacks:

• Nothing is done to prevent misbehaving (and thereby possibly malicious) nodes from being used for routing, as long as they have the required key

• If a malicious node somehow retrieves the required key the protocol is still open for all kinds of attacks mentioned previously

There is one other main drawback to using SAR and that is the excessive encrypting and decrypting required at each hop. Because we are dealing with mobile environments the extra processing leading to increased power consumption can be a problem, depending on the kind of mobile devices being used.

SAR is intended for the managed-open environment as it requires some sort of key distribution system in order to distribute the trust level keys to the correct devices.

b) Secure Routing Protocol, SRP

Secure Routing Protocol, SRP, is another protocol extension that can be applied to any of the most commonly used protocols today. The basic idea of SRP is to set up a security association (SA) between the source and the destination node. The SA is usually set up by negotiating a shared key based on the other party's public key, and after that the key can be used to encrypt and decrypt the messages. The routing path is always sent along with the packets, unencrypted though (since none of the intermediate nodes have knowledge of the shared key), thus exposing network infrastructure information to potential attackers. In fact one of the main security issues in SRP is that it has no defense against the "invisible node" attack that simply puts itself (and possibly a large number of other invisible nodes) somewhere along the message path without adding itself to the path, thereby causing potentially big problems as far as routing goes.

c) The Selfish Node

The selfish node is based on one of Darwin's theories of evolution within birds, where birds are divided into suckers (always helping others), cheats (never helping, always receiving help) and grudgers (help those that help them). The theory states that eventually the suckers die first, and then the cheats (since the grudgers won't help them) and the grudgers will reign. This concept is moved to the open environment ad hoc networks in order to help avoid maliciously behaving nodes.

The open environment poses quite a few new threats to ad-hoc networks. Among others, it is very difficult to recognize a malicious node using certificates since the idea of this kind of environment is that different devices, presumably from very different locations and owners, cooperate to create a functioning network. Since the main goal of such a network is high throughput the simplest and therefore most probable form of attack targeting the main goal is a DOS-attack, and this is what they're trying to prevent.

Using suitably sized cost and profit to routing and forwarding the goal is to more or less isolate misbehaving (possibly malicious) nodes. The following components are used in order to try and keep network throughput at a maximum:

• The Monitor

This component acts as a sort of a "neighborhood-watch", where nodes try to detect bad behavior in nodes in their vicinity. Bad behavior that can be detected can be unusually high routing traffic (possible "Black Hole" attack), unusually frequent routing updates ("flooding") and more. Of course, reasonable thresholds must be used in order for this to work. When bad behavior is detected an alarm signal is sent to the reputation system.

• The Reputation System

This is basically a rating of nodes and what their reputation is. Depending on reported alarms and alarms experienced by the node itself different nodes are rated differently. This component can also use a rumor spreading system to inform other nodes of bad behaving nodes reputation. This way a malicious node will quickly become "notorious" among the other nodes.

• The Path Manager

The path manager is responsible for taking the appropriate changes in routing tables as alarms and reputations changes in the system, deletion of malicious behaving nodes from routing tables for instance.

• The Trust Manager

The trust manager maintains a list of nodes and how much they are trusted. When an alarm is received depending on how trustworthy the reporting node is, different actions can be taken, since we of course don't want to leave ourselves open for attacks where malicious nodes tries to ban other nodes by sending false alarms.

Each of these components exists within each node and they all help to keep the network alive. The result is a network that in a sense learns that some of the nodes are malicious and therefore isolate them.

Indeed this is a very different approach then the other mentioned systems but keep in mind that this is the only one really intended for the open environment, with nodes of unknown origin cooperating to achieve maximum network throughput. That is why it is focused on different kinds of DOS attacks and not concentrating on encrypting traffic and such. Also note that in the open environment no use of existing infrastructure is to be used, which leaves the previously mentioned systems useless since they more or less all require existing infrastructure (i.e. certificate servers).

2) Secure protocols

These are protocols designed from the ground up to provide ad-hoc networks with all the required features described earlier.

a) Authenticated Routing for Ad-hoc Networks, ARAN

ARAN is a protocol designed to provide secure communications in managed-open environments. Like SAR it makes use of existing infrastructure in the form of certificate servers. The protocol has two phases, authentication and transmission.

1. Authentication

The goal of the first phase is to make sure that a secure path from the source node A and the destination node B can be established. The phase requires that each node has received a certificate from a trusted certificate server. The certificate contains a node's IP number, public key as well as the time of issuing and expiration.

Node A broadcasts a signed (using A's key) route discovery packet (RDP) to all its neighboring nodes in order to find a route to B. Each node that receives the RDP for the

first time removes any other intermediate(not A) node's signature, signs the RDP using its own key and broadcasts it to all its neighboring nodes, saving a route pair (A,B) in its routing table. This continues until node B eventually receives the packet. Node B then sends a reply packet containing its own certificate and signed using its key, the packet is sent along the reverse path (each intermediate node sends it back to where the original RDP came from). When A receives the REP packet, it's checks that the signature is correct and stores node B's certificate to use in the next phase.
The procedure does ensure loop freedom as well as makes sure that B really is B using the certificates (providing of course that the certificate server has not been compromised). One of the downsides to this procedure is that each node has to store the source-destination routing pairs instead of just routing based on destination which is used in other protocols.

2. Transmission

A now needs to discover the shortest path to B and therefore sends a "Shortest Path Confirmation", SPC, packet to all its neighbors, encrypted using B's public key. Each successive intermediate node encrypts the message again using B's public key, including its own certificate, and forwards it to its neighbors. When B eventually receives the SPC packet it checks all of the signatures and replies to the first SPC received, as well as all other SPCs having a shorter recorded path (the path is recorded in the encrypted keys). B then sends a "Recorded Shortest Path", RSP, packet back to A, including the path to use in the packet. A can safely verify that it comes from B and that it corresponds to the original SPC sent. This way A now has a shortest, secure path to B to transmit data over.

Since at all forwarding the packet is re-encrypted using B's public key, only B is able to discover the actual route taken. This way any spoofing attacks or other attempts to misdirect the packets will fail since the malicious nodes first would have to crack the encryption. Only using B's private key would that be possible. Hence, the so called "invisible node-attack" is also prevented using this protocol.
One of the main issues using ARAN is the required certificate server, which means that the integrity of that server is vital. This is by design though and as it is intended for an managed-open environment it shouldn't be considered a big issue.

b) Secure Position Aided Ad hoc Routing, SPAAR

The Secure Position Aided Ad hoc Routing, SPAAR, protocol was developed with the classical managed-hostile environment in mind, thus meant to provide a very high level of security, sometimes at the cost of performance. Among other things, SPAAR also requires that each device use a GPS locator to determine its position, although some leeway is given to nodes using a so-called "locator-proxy" if absolute security is not required.

The certificate system is similar to ARAN in that a combination of a public key and the public key of the certificate server is used, although in SPAAR a third key is also generated, a group neighborhood key that is used to decrypt Route Request packets, RREQ. In SPAAR packets are only accepted between neighboring nodes one hop away from each other, this is to avoid the "invisible node-attack". Besides certificates nodes also use the location of the other nodes when attempting to communicate, a maximum distance N is set that decides how far away a node can be and still be called a neighbor.

The basic transmission procedure is quite similar to ARAN, although the group neighborhood key is used for encryption in order to ensure one-hop communication only. Since all nodes also have information on their location they only forward RREQs if their position is closer to the destination position. In the destination node reply the location and velocity-vector of the destination is returned, this is necessary since the source node needs to know the approximate location of the destination in order for the routing to be efficient.

SPAAR may seem a bit extreme, using multiple keys and GPS location-dependent routing. Considering the nature of the managed-hostile environment this is not very strange. In the situations this environment presents, finding the geographically shortest path can be at least as important as finding the fastest path, whether its in a battle field or a disaster area. Also, it reveals no information on the network layout to any non-authorized nodes, something which also can be essential when relay stations are secret.

The only real security problem currently discovered in SPAAR is once again the usage of the certificate server and the extreme need to keep this server uncompromised. Also, issues still exist with compromised nodes already having valid certificates.

Conclusion:

Thus Ad-hoc networks are prove to various kinds of vulnerable attacks since they are dynamic, wireless and infrastructure less network. Besides all these hazards there are the presence of security routing protocols which make them more secure and error-free networks there by admiring the ultimate aim of Ad-hoc networks that is the accomplishment of instant network regardless of the types of nodes or type of environments that is prevailing which is described in this paper.

References:

1. Intrusion Detection in Wireless Ad-hoc Networks, Yongguang Zhang, Wenke Lee, 2000, http://www.wins.hrl.com/people/ygz/papers/mobicom00.pdf
2. Securing Ad-hoc Networks, L. Zhou, Z.J.Haas, 1999, Cornell University, http://www.cs.cornell.edu/home/ldzhou/adhoc.pdf
3. A Secure Routing Protocol for Ad Hoc Networks, Bridget Dahill, Brian Neil, Elizabeth Royer, Clay Shields, 2000, http://www.cs.umn.edu/research/mobile/seminar/SUMMER03/WNfiles/aran.icnp02.pdf
4. Security-Aware Ad-Hoc Routing for Wireless Networks, Seung Yi, Prasad Naldurg, Robin Kravets, 2001, University of Illinois, http://www-old.cs.uiuc.edu/Dienst/Repository/2.0/Body/ncstrl.uiuc_cs/UIUCDCS-R-2001-2241/pdf
5. Mitigating Routing Misbehaviour in Ad Hoc Networks, S. Marti, T.J. Giuli, K Lai and M. Baker, 2000, Stanford University, http://mosquitonet.stanford.edu/~laik/projects/adhoc/mitigating.pdf