2005 Asia-Pacific Conference on Communications, Perth, Western Australia, 3 - 5 October 2005.

Secure Routing with the AODV Protocol

Asad Amir Pirzada and Chris McDonald
School of Computer Science & Software Engineering,
The University of Western Australia
35 Stirling Highway, Crawley, W.A. 6009, Australia.
Email: {pirzada,chris}@csse.uwa.edu.au

Abstract— **Ad-hoc networks, due to their improvised nature, are frequently established in insecure environments, which makes them susceptible to attacks. These attacks are launched by participating malicious nodes against different network services. Routing protocols, which act as the binding force in these networks, are a common target of these nodes. Ad-hoc On-Demand Distance Vector (AODV) is one of the widely used routing protocols that is currently undergoing extensive research and development. AODV is based on distance vector routing, but the updates are shared not on a periodic basis but on an as per requirement basis. The control packets contain a hop-count and sequence number field that identifies the freshness of routing updates. As these fields are mutable, it creates a potential vulnerability that is frequently exploited by malicious nodes to advertise better routes. Similarly, transmission of routing updates in clear text also discloses vital information about the network topology, which is again a potential security hazard. In this paper we present a novel and pragmatic scheme for securing the Ad-hoc On-Demand Distance Vector routing protocol that protects against a number of attacks carried out against mobile ad-hoc wireless networks.**

Keywords: Trust, Security, Ad-hoc, Networks, Protocols

I. Introduction

Mobile ad-hoc wireless networks hold the promise of the future, with the capability to establish networks at anytime, anywhere. These networks don't rely on extraneous hardware which makes them an ideal candidate for rescue and emergency operations. These networks are built, operated and maintained by its constituent wireless nodes. These nodes generally have a limited transmission range and so each node seeks the assistance of its neighbouring nodes in forwarding packets. In order, to establish routes between nodes, which are farther than a single hop, specially configured routing protocol are engaged. The unique feature of these protocols is their ability to trace routes in spite of a dynamic topology.

These protocols can by far and large be categorised into two main types: Reactive and Proactive [1]. The nodes in an ad-hoc network generally have limited battery power and so active routing protocols endeavour to save upon the same, by discovering routes only when they are essentially required. In contrast, proactive routing protocols establish and maintain routes at all instants of time, so as to avoid the latency that occurs during new route discoveries. Both types of routing protocols require persistent cooperative behaviour, with intermediate nodes primarily contributing to the route development. Similarly each node, which practically acts like a mobile router [2], has absolute control over the data that passes through it. In essence, the membership of any ad-hoc networks indisputably calls for sustained depiction of benevolent behaviour by all participating nodes.

However, this is more than often difficult to achieve in an open environment and so these networks are frequently attacked by malicious nodes, which may originate internally or join externally. Two kinds of attacks can be launched against ad-hoc networks [3], Passive and Active. In passive attacks the attacker does not disturb the routing protocol. It only eavesdrops upon the routing traffic and endeavours to extract valuable information like node hierarchy and network topology from it. In active attacks, malicious nodes can disturb the correct functioning of a routing protocol by modifying routing information, by fabricating false routing information, and by impersonating other nodes [4].

Cryptographic mechanisms are commonly used to protect routing protocols by enforcing mutual trust relationships among the wireless nodes [5]. Security in mobile ad-hoc wireless networks is a two-fold problem. One is the security of the routing protocols that enable the nodes to communicate with each other and the second is the protection of the data that traverses the network on routes established by the routing protocols.

In this paper, after Introduction in Section I, we describe some recent secure routing protocols for ad-hoc networks in Section II, which have been developed to counter known attacks. In Section III we propose a scheme for securing the Ad-hoc On-Demand Distance Vector routing protocol. A security analysis of the proposed scheme is presented in Section IV with concluding remarks in Section V.

II. Previous Work

To protect an ad-hoc network from attacks a routing protocol must fulfil a set of requirements [4] to ensure that the discovered path from source to destination functions properly in the presence of malicious nodes. These are:

1) Authorized nodes should perform route computation and discovery,
2) Minimal exposure of network topology,
3) Detection of spoofed routing messages,
4) Detection of fabricated routing messages,
5) Detection of altered routing messages,
6) Avoiding formation of routing loops, and
7) Prevent redirection of routes from shortest paths.

A number of secure routing protocols [6] have been recently developed that conform to most of the requirements. These protocols employ a variety of cryptographic tools for protecting the vulnerabilities in different routing protocols . However, these protocols have been developed as a practical response to specific problems that arose due to attacks on ad-hoc network routing protocols. Consequently, these protocols only cover a subset of all possible threats and are not flexible enough to be integrated with each other. Some of the recent secure routing protocols are explained in the following sub-sections.

A. ARAN

The Authenticated Routing for Ad-hoc Networks (ARAN) [4] secure routing protocol is an on-demand routing protocol that identifies and shields against malevolent actions by malicious nodes in the ad-hoc network environment. ARAN relies on the use of digital certificates and can successfully operate in the managed-open scenario where no network infrastructure is pre-deployed, but a small amount of prior security coordination is expected. ARAN provides authentication, message integrity and non-repudiation in ad-hoc networks by using a preliminary certification process that is followed by a route instantiation process that guarantees end-to-end provisioning of security services. ARAN requires the use of a trusted certificate server. All nodes are supposed to keep fresh certificates with the trusted server and should know the server's public key. Prior to entering the ad-hoc network, each node has to apply for a certificate that is signed by the certificate server. The certificate contains the IP address of the node, its public key, a timestamp of when the certificate was generated and a time at which the certificate expires, along with the signature by the certificate server. ARAN accomplishes the discovery of routes by a broadcast route discovery message from a source node, which is replied to in a unicast manner by the destination node. All the routing messages are authenticated at every hop from the source to the destination, as well as on the reverse path from destination to source

B. SAODV

The Secure Ad-hoc On-Demand Distance Vector (SAODV) [7] is an extension of the AODV routing protocol. It can be used to protect the route discovery mechanism of AODV by providing security features like integrity, authentication and non-repudiation. The protocol operates mainly by using new extension messages with the AODV protocol. In these extension messages there is a signature produced by digesting the AODV packet using the private key of the original sender of the routing message. The Secure-AODV scheme is based on the assumption that each node possesses certified public keys of all network nodes. Ownership of certified public keys enables intermediate nodes to authenticate all in-transit routing packets. The originator of a routing control packet appends its RSA signature and the last element of a hash chain to the routing packets. As the packets traverse the network, intermediate nodes cryptographically authenticate the signature and the hash value. The intermediate nodes generate the k^{th} element of the hash chain, with k being the number of traversed hops, and place it in the packet. The route replies are supplied either by the destination or intermediate nodes having an active route to the required destination.

III. Securing the Ad-hoc On-Demand Distance Vector Routing Protocol

A. AODV Protocol

Ad-hoc On-Demand Distance Vector (AODV) [8] is inherently a distance vector routing protocol that has been optimised for ad-hoc wireless networks. It is an on demand protocol as it finds the routes only when required and is hence also reactive in nature. AODV borrows basic route establishment and maintenance mechanisms from the DSR protocol and hop-to-hop routing vectors from the DSDV protocol. To avoid the problem of routing loops, AODV makes extensive use of sequence numbers in control packets. When a source node intends communicating with a destination node whose route is not known, it broadcasts a `ROUTE REQUEST` packet. Each `ROUTE REQUEST` packet contains an ID, source and the destination node IP addresses and sequence numbers together with a hop count and control flags. The ID field uniquely identifies the `ROUTE REQUEST` packet; the sequence numbers inform regarding the freshness of control packets and the hop-count maintains the number of nodes between the source and the destination. Each recipient of the `ROUTE REQUEST` packet that has not seen the Source IP and ID pair or doesn't maintain a fresher (larger sequence number) route to the destination rebroadcasts the same packet after incrementing the hop-count. Such intermediate nodes also create and preserve a `REVERSE ROUTE` to the source node for a certain interval of time. When the `ROUTE REQUEST` packet reaches the destination node or any node that has a fresher route to the destination a `ROUTE REPLY` packet is generated and unicasted back to the source of the `ROUTE REQUEST` packet. Each `ROUTE REPLY` packet contains the destination sequence number, the source and the destination IP addresses, route lifetime together with a hop count and control flags. Each intermediate node that receives the `ROUTE REPLY` packet, increments the hop-count, establishes a `FORWARD ROUTE` to the source of the packet and transmits the packet on the `REVERSE ROUTE`. For preserving connectivity information, AODV makes use of periodic `HELLO` messages to detect link breakages to nodes that it considers as its immediate neighbours. In case a link break is detected for a next hop of an active route a `ROUTE ERROR` message is sent to its active neighbours that were using that particular route.

The major vulnerabilities present in the AODV protocol are:

1) Deceptive incrementing of Sequence Numbers: Destination Sequence numbers determine the freshness of a route. The destination sequence numbers maintained by different nodes are only updated when a newer control packet is received with a higher sequence number. Normally the destination sequence numbers received via control packets cannot be greater than the previous value held by the node plus one [9]. However, malicious nodes may increase this number so as to advertise fresher routes towards a particular destination. If this difference is equal or larger than two then there is a high probability that the network may be under a modification attack.

2) Deceptive decrementing of Hop Count: AODV prefers route freshness over route length. In that, a node prefers a control packet with a larger destination sequence and hop count over a control packet with a smaller destination sequence and hop count. However, if the destination sequence numbers are the same then the route with the least hop count is given preference. Malicious nodes frequently exploit this mechanism in order to generate fallacious routes that portray minimal hop-counts.

B. Secure AODV Routing Protocol

Securing the AODV protocol can be divided into the following three broad categories:

1) Key Exchange
2) Secure Routing
3) Data Protection

1) Key Exchange: Most of the current key exchange protocols are dependent upon a central trust authority for initial authentication. A variant of the central trust authority is the Distributed Public-Key Model [10] that makes use of threshold cryptography to distribute the private key of the Certification Authority (CA) over a number of servers. Whatever the case may be, the requirement of a central trust authority in such a dynamic environment is considered both impractical and unsafe, as such an entity may not always be accessible and it also creates a single point of failure. Similarly, key exchange using a Key Distribution Server [11] creates a similar set of problems. We suggest that all nodes, before entering a network, procure a one-time public and private key pair from the CA along with the CA's public key. After this, the nodes can negotiate session keys among each other, without any reliance on the CA, using any suitable key exchange protocol for ad-hoc networks [12]. These session keys are used for securing the routing process and subsequently the data flow. To avoid multiple peer-to-peer encryptions during broadcast or multicast operations, a group session key may be established between immediate neighbours using a suitable Group Keying Protocol [12]. This mechanism absolves the ad-hoc network of superfluous requirements and provides necessary elements to secure both routing and data in presence of malicious nodes by providing security services like authentication, non-repudiation, confidentiality and integrity.

2) Secure Routing: Ad-hoc On-Demand Distance Vector routing protocol operates at the third layer of the TCP/IP protocol suite using UDP port 654. The source node that requires a route to a destination broadcasts a `ROUTE REQUEST` packet, each intermediate recipient node retransmits the packet, if not a duplicate, and the final destination unicasts a `ROUTE REPLY` packet back to the original sender. For route maintenance it uses `ROUTE ERROR` packets that inform active users of route failures. The `ROUTE REQUEST` and `ROUTE REPLY` packets are usually modified by the intermediate nodes so as to add necessary routing information to these packets. The core security related problems linked to ad-hoc networks originate due to the route development by the intermediate nodes. It is therefore, imperative that only authorised nodes are allowed to update routing packets and malicious nodes be avoided at all costs. To restrict modification of routing packets by intermediate nodes, we recommend peer-to-peer symmetric encryption of all routing information. All routing control packets between nodes are first encrypted and then transmitted. The sequence of steps, for route discovery and route maintenance, is as follows:

Route Request:

1) Any Node 'x' desiring to establish communication with another Node 'y' first establishes a group session key K_x with its immediate neighbours (nodes that are a single hop away) as shown in Figure 1.
2) It then creates the `ROUTE REQUEST` packet as per the routing protocol specifications shown in Figure 2.

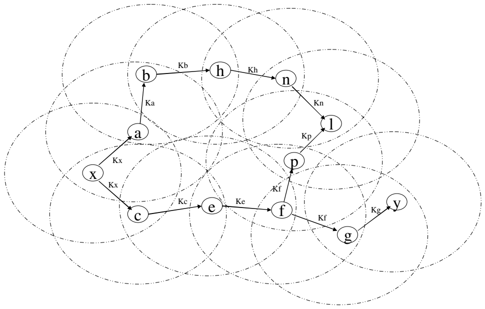

Fig. 1. Point-to-Point Establishment of Secure Routes

3) The `ROUTE REQUEST` packet is then encrypted using the group session key K_x and broadcasted.
4) All intermediate recipient nodes that share the same group session key decrypt the `ROUTE REQUEST` packet and, if required, modify it according to the routing protocol specifications.
5) The intermediate nodes that do not possess group session keys with their immediate neighbours, initiate the group session key exchange protocol.
6) After establishing the group session key, the intermediate nodes encrypt the `ROUTE REQUEST` packet using the new session key and rebroadcast the packet.
7) Steps 4 to 6 are followed until the final destination Node 'y' receives the packet.

Type	J	R	G	D	U	Reserved	Hop Count
RREQ ID							
Destination IP Address							
Destination Sequence Number							
Originator IP Address							
Originator Sequence Number							

Fig. 2. Route Request (RREQ) Message Format

Route Reply:

1) In response to the `ROUTE REPLY` packet Node 'y' creates a `ROUTE REPLY` packet as per the routing protocol specifications shown in Figure 3.

Type	R	A	Reserved	Prefix Size	Hop Count
Destination IP address					
Destination Sequence Number					
Originator IP address					
Lifetime					

Fig. 3. Route Reply (RREP) Message Format

2) The ROUTE REPLY packet is encrypted using the last group session key (K_g in this case) that was used to decrypt the received ROUTE REQUEST packet and is unicast back to the original sender.
3) If any of the intermediate nodes has moved out of the wireless range a new group session key is established.
4) All recipient nodes that share the forward group session key decrypt the ROUTE REPLY packet and, if required, modify it according to the routing protocol specifications.
5) The ROUTE REPLY packet is then again encrypted using the backward group session key and unicast towards Node 'x'.
6) Steps 4 and 5 are repeated until the packet is received by Node 'x'.

To avoid key synchronisation problems it is recommended that each node maintain a table indexed by Node ID as the primary key along with associated group members and session keys as shown in Figure 4. The table also helps establish secure routes with other nodes with which a chain can be established using the available session keys. A secure chain is highlighted in the figure between Node 'x' and 'y'.

		Destination											
Source	ID	a	b	c	e	f	g	h	l	n	p	x	y
	a		K_a									K_x	
	b	K_a						K_b					
	c				K_c							K_x	
	e			K_c		K_e							
	f				K_e		K_f				K_f		
	g					K_f							K_g
	h		K_b							K_h			
	l									K_n	K_p		
	n							K_h	K_n				
	p					K_f			K_p				
	x	K_x		K_x									
	y						K_g						

Fig. 4. Session Key Table

Route Maintenance: In a mobile ad-hoc network, established routes may be broken due to a variety of reasons. However, the underlying routing protocol takes care of such events by either gratuitously repairing them or sending a ROUTE ERROR packet to inform the nodes currently using the route. All messages associated with route maintenance also need to be authenticated and protected from eavesdropping. If a packet is received for an inoperative route the recipient node takes the following steps:

1) The node detecting the broken link creates a ROUTE ERROR packet as per the routing protocol specifications shown in Figure 5.
2) This packet is then encrypted using a group session key in the direction of the recipient node using the Session Key Table and is multicast back to the recipients.
3) If any of the intermediate nodes has moved out of the wireless range a new group session key is established.
4) All recipient nodes that share the group session key decrypt the ROUTE ERROR packet, and if required, modify it according to the routing protocol specifications.
5) The ROUTE ERROR packet is then again encrypted using the group session key and is multicast back to the recipients.
6) Steps 4 and 5 are repeated until the intended recipients receive the ROUTE ERROR packet.

Type	N	Reserved	Dest Count
Unreachable Destination IP Address			
Unreachable Destination Sequence Number			
Additional Unreachable Destination IP Addresses (if needed)			
Additional Unreachable Destination Sequence Numbers (if needed)			

Fig. 5. Route Error (RERR) Message Format

3) Data Protection: Once protected routes have been established, secure data transfer is relatively straightforward. To ensure connection confidentiality a source node adopts the following steps:

1) Any Node 'x' desiring to establish an end-to-end secure data channel, first establishes a session key K_{xy} with the intended Node 'y' using the key exchange protocol as shown in Figure 6.

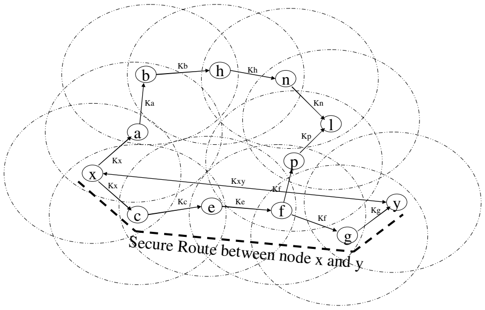

Fig. 6. End-to-End Establishment of Secure Routes

2) It then symmetrically encrypts the data packet using the session key K_{xy} and transmits it over the secure route.
3) The intermediate nodes simply forward the packet in the intended direction.
4) When the encrypted data packet reaches the destination it is decrypted using the session key K_{xy}.
5) Steps 2 to 4 are followed for all further data communication.

IV. Security Analysis

In this section we discuss how the presented security scheme defies possible attacks in an ad-hoc network. As discussed earlier, the basis of a security infrastructure is primarily dependent on the initial key exchange providing authentication. Other security services like confidentiality, integrity and non-repudiation all rely on the accuracy of the authentication service. Key revocation, being an important issue has not been addressed in the scope of this paper, primarily because it requires the presence of an omnipresent, and often omniscient, trust authority, which we have already deemed inappropriate for such a dynamic environment. We now discuss how this scheme satisfies the seven requirements of any secure routing protocol:

A. Authorised nodes to perform route computation and discovery

The authentication and key exchange protocol ensures that only authorised nodes are able to perform the route discovery. As the routing control packets are encrypted and authenticated by each forwarding node, malicious nodes will not be able to create fallacious routing packets.

B. Minimal exposure of network topology

As all routing information is encrypted between nodes, an adversary will gain no information regarding the network topology from passive eavesdropping.

C. Detection of spoofed routing messages

Spoofing of either the MAC or IP addresses does not provide any benefit to the adversary until the time the authentication protocol is assumed to be secure. As the initial authentication links a number of identities to each node's private key, the spoofing node will have to create a similar private key prior to launching any attack.

D. Detection of fabricated routing messages

Malicious nodes cannot inject fabricated routing messages into the network as each routing packet is secured through an encryption key, which provides the benefit of confidentiality, authentication and integrity at the same time. To fabricate a routing message the session key needs to be compromised, which is not possible until the time the key exchange protocol is assumed to be secure.

E. Detection of altered routing messages

Routing messages are relayed between the nodes in an unintelligible format. If the symmetric cipher also provides the integrity then the alteration of routing messages is virtually impossible. Addition of a keyed hash for better integrity checking may be considered only after a cost-benefit analysis.

F. Avoiding formation of routing loops

The proposed scheme ensures that routing loops cannot be formed through malicious action. Routing loops usually occur if a malicious node is able to spoof, alter or fabricate legitimate routing packets.

G. Prevent redirection of routes from shortest paths

Shortest paths are created usually by decrementing the number of addresses in the source routing protocol. The scheme is designed in such a manner that routing packets are only accepted from authenticated immediate neighbours. This ensures that an adversary cannot inject such routing packets unless an authorised node first authenticates it.

V. Conclusion

In this paper we have presented a scheme for securing the Ad-hoc On-Demand Distance Vector routing protocol used in mobile ad-hoc wireless networks. The secure AODV protocol provides requisite measures for protection of route discovery and transfer of data. These measures can be exercised independently without a central trust authority with nodes negotiating session keys independently. Nodes are, however, required to register themselves once with a Certification Authority, prior to joining a network. The scheme is based upon point-to-point and end-to-end encryption using symmetric key-based mechanisms. Nodes desiring secure communication, execute any standard authentication and key exchange protocol to acquire session keys. These keys are subsequently used in point-to-point encryption for route discovery and end-to-end encryption for data packets. Malicious nodes trying to launch passive or active attacks against the network are thwarted through efficient key verification mechanisms and a multi-layered enciphering scheme. To highlight its viability we have discussed its resistance to a number of attacks specific to ad-hoc networks.

Acknowledgements

This work was supported by the Australian International Postgraduate Research Scholarship and the University of Western Australia Postgraduate Award.

References

[1] E. M. Royer and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications Magazine*, vol. 6, no. 2, pp. 46–55, 1999.

[2] S. Corson and J. Macker, "Mobile ad-hoc networking (manet): Routing protocol performance issues and evaluation considerations," *IETF MANET, RFC 2501*, 1999.

[3] Y. C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," *Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 12–23, 2002.

[4] B. Dahill, B. N. Levine, E. Royer, and C. Shields, "A secure routing protocol for ad hoc networks," *Proceedings of the International Conference on Network Protocols (ICNP)*, pp. 78–87, 2002.

[5] A. A. Pirzada and C. McDonald, "Establishing trust in pure ad-hoc networks," *Proceedings of the 27th Australasian Computer Science Conference (ACSC)*, vol. 26, no. 1, pp. 47–54, 2004.

[6] ——, "Secure routing protocols for mobile ad-hoc wireless networks," in *Advanced Wired and Wireless Networks*, T. A. Wysocki, A. Dadej, and B. J. Wysocki, Eds. Springer, 2004.

[7] M. G. Zapata, "Secure ad hoc on-demand distance vector (saodv) routing," *IETF MANET, Internet Draft (work in progress)*, 2001.

[8] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (aodv) routing," *IETF RFC 3591*, 2003.

[9] W. Wang, Y. Lu, and B. Bhargava, "On vulnerability and protection of ad hoc on-demand distance vector protocol," *Proceedings of the International Conference on Telecommunication (ICT)*, pp. 375–382, 2003.

[10] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network Magazine*, vol. 13, no. 6, pp. 24–30, 1999.

[11] A. A. Pirzada and C. McDonald, "Kerberos assisted authentication in mobile ad-hoc networks," *Proceedings of the 27th Australasian Computer Science Conference (ACSC)*, vol. 26, no. 1, pp. 41–46, 2004.

[12] D. Carman, P. Kruus, and B. Matt, "Constraints and approaches for distributed sensor network security," Technical Report 00-010, NAI Labs, 2000.