Security-Aware Ad-Hoc Routing for Wireless Networks

Seung Yi Prasad Naldurg Robin Kravets

Department of Computer Science
University of Illinois at Urbana-Champaign
1304 West Springfield Avenue, Urbana, IL 61801-2987 USA

{*seungyi, naldurg, rhk*}*@cs.uiuc.edu*

Report No. UIUCDCS-R-2001-2241, UILU-ENG-2001-1748

August, 2001

Security-Aware Ad-Hoc Routing for Wireless Networks

1 Introduction

Wireless ad hoc networks have been proposed to support dynamic scenarios where no wired infrastructure exist. Ad hoc environments introduce two main problems not commonly faced by traditional fixed-network routing protocols. These are the lack of fixed infrastructure support and the frequent changes to network topology. At the physical level, wireless channels offer poor protection to protocol packets and are susceptible to signal interference, jamming, eavesdropping, and distortion. Adding additional processing (e.g., for reliability), and intelligence (error-correcting codes, frequency hopping, etc.) at the physical and MAC layers, overcomes some of these limitations. At the network layer, most ad hoc routing protocols are cooperative by nature [6], and rely on implicit trust-your-neighbor relationships to route packets among participating nodes. This naive trust model allows malicious nodes to paralyze an ad hoc network by inserting erroneous routing updates, replaying old routing information, changing routing updates, or advertising incorrect routing information [25, 15]. While these attacks are possible in fixed networks as well, the nature of the ad hoc environment magnifies their effects, and makes their detection difficult [23].

The characteristics of an ad hoc network demand new metrics for routing. Traditionally, distance (measured in hops) is used as the metric in most ad hoc route-discovery algorithms (e.g., AODV, DSR, ZRP, TORA etc.). The use of other metrics (e.g., geographic location, signal stability, power, load on nodes etc. [14, 22, 17]) can improve the quality and the relevance of the routes discovered for particular applications and configurations. Along these lines, we explore the use of different security attributes to improve the quality of the security of an ad-hoc route. In this paper, we present "Security-Aware ad-hoc routing (SAR)", an approach to routing that incorporates security levels of nodes into traditional routing metrics. Our goal is to characterize and explicitly represent the trust values and trust relationships associated with ad hoc nodes and use these values to make routing decisions. We quantify the notion of trust and represent the trust relationships explicitly by defining a suitable hierarchy of trust values.

Ensuring that data is routed through a secure route composed of trusted nodes is only one half of the problem. The second issue is the security of the information in the routing protocol messages. Since both data and control messages use the same wireless transmission medium, routing protocol messages can be altered to change routing behavior. For example, if a routing protocol update can be subverted, and the message altered in transit, no amount of security on data packets can mitigate routing misbehavior. In this paper, we also analyze the security of ad hoc routing algorithms with respect to the protection associated with the transmission of routing messages. One of the reasons for exposing security attributes at the routing level is to prevent attacks on the routing protocol itself, and thereby secure a fundamental building block of the ad hoc network infrastructure. We identify the attributes of a secure route and define appropriate metrics to quantify the "level of security" associated with the protocol messages. These metrics are adapted from their equivalents in security of wired routing protocols [16, 18, 2].

The trust value of a node and the security attributes of a route are intimately connected in our framework to provide a unified view of security. We introduce the notion of an integrated security metric that is a combination of the security attributes and trust levels. We augment existing ad hoc routing algorithms with this integrated metric to influence the route discovery and route maintenance behavior. Our route discovery mechanism finds nodes that match particular

security attributes and trust levels. Only nodes that provide the required level of security can generate or propagate route requests, updates, or replies. This means that our protocol generates fewer routing protocol messages. Security policies can be encoded into the attributes to enable policy-based secure routing.

Introducing security into the routing protocol has an associated performance penalty. Rather than propose a specific solution, we develop a generalized framework with open feedback and explicit representation of attributes and choices. This allows users to change the security attributes dynamically and renegotiate for alternative routes, based on (but not limited to) the cost-benefit analysis of the performance penalties versus the protection offered in the given situation. For example, when the route discovery mechanism fails to find a route with the requested security attributes, the protocol initiator can choose to send another route discovery request with modified attributes to find a route with different security guarantees. Our mechanism is transparent to applications, though security aware applications can take advantage of the feedback feature in our framework to adapt to situational needs and constraints.

The rest of the paper is organized as follows: In Section 2, we present our motivation with a representative example, expand on the characteristics of an ad hoc network that make it vulnerable to routing attacks and briefly describe a threat model. In Section 3, we present our generalized SAR protocol for quantifiable secure route discovery, update, and propagation with trust levels and security attributes as metrics. This section includes related research, a description of the traditional definitions and metrics of routing protocol security, and outlines a mechanism to quantify and measure the protection associated with particular routing protocol incarnations. In Section 4, we revisit our threat model, develop an attack classification and validate our protocol against this model. Section 5 describes our experimental test bed and our modifications to AODV to enable security aware routing. Section 6 presents our ns2 simulation results in detail. Finally, Section 7 presents our conclusions.

2 Motivation

Communication among nodes in an ad hoc network is accomplished with support from the routing protocol. While the dynamics of these protocols have been well researched, the security issues and concerns have not been addressed in depth. In this section we exemplify the need for security awareness in an ad hoc network at the routing level with a battlefield communication scenario. Next, we identify characteristics of an ad hoc network that make it susceptible to attacks on its routing protocols, and describe an informal threat model. In the next section, we describe the SAR protocol itself.

2.1 Example

We present an example scenario where finding a route with specific security attributes or trust levels is more relevant than finding the shortest route (or any route) between two nodes. We focus on a high-risk ad hoc network; wireless communication devices in a battlefield, where malicious adversaries can intercept and alter mission critical information.

In Figure 1, two generals establish a route to communicate among themselves, using a generic on-demand ad-hoc routing protocol. During the mission, the generals detect that some of the privates have defected. The generals decide that they can only trust nodes owned by officers to route their packets. Relaying these messages using potentially compromised nodes can leak information to untrusted entities and jeopardize the mission. Even if the generals encrypt the

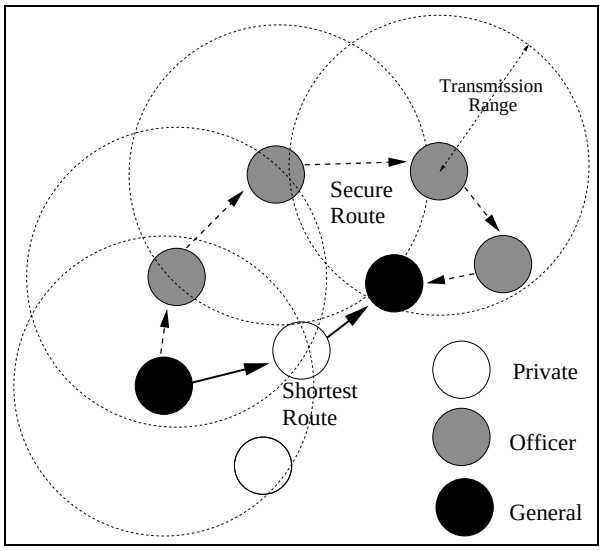

Figure 1: Security-aware Routing - Motivation

information flowing between them, the fact that they are communicating may disclose that a strike is imminent. Another threat could be that traitors may be able to store the messages or send them to enemy nodes for cryptanalysis. Using SAR, the generals can route around the problem nodes and establish an alternate route with greater security guarantees. The sending general's route discovery protocol embeds the rank of the node as a metric in its negotiation and tries to establish a route that avoids all privates. If the protocol can find the route, as shown in the figure, a session passing through only the officers is set up. If the protocol fails to find a route with the required security attributes or "quality of protection", it sends a notification to the sender and allows re-negotiation. Based on the feedback from the protocol, for example, the generals may decide to set up a route that can support 128-bit encryption, knowing that the privates cannot cryptanalyze or transmit the messages very far with their smaller devices.

From this example, we observe that the senders or protocol initiators can make informed decisions about the "quality of protection" available to their data packets by embedding security attributes into the route discovery protocol itself. Furthermore, the quality of protection offered by the route directly affects the security of the data packets exchanged between the nodes on a particular route. Route updates and route propagation messages are also protected by this technique. In the next subsection, we enumerate characteristics of the ad hoc network environment that make them vulnerable to attacks. To strengthen our motivation to include security as a fundamental attribute or metric in ad hoc routing protocols, we describe an informal threat model. The model enumerates the vulnerabilities and threats that expose the communication of routing protocol packets among nodes in an ad hoc network, to malicious attackers.

2.2 Ad hoc Network Characteristics

An ad hoc network has many characteristics that contrast sharply with fixed networks or last-hop wireless networks. First, there is no infrastructure support. All routers are mobile and can communicate with each other only when they are in transmission range. Second, ad hoc wireless nodes are resource constrained, with limited processing and memory capacity, and are usually powered with batteries. Finally, the communication medium in an ad hoc wireless network, i.e.,

radio waves, infrared, etc., can be easily eavesdropped. Hostile environments like battlefields or commando rescue operations are some of the important target application areas for ad-hoc wireless networks.

Ad hoc wireless routing protocols assume that the mobile nodes are cooperating with each other to route a packet from the source to their destinations. Routing protocol packets carry important control information that governs the behavior of data transmission in the ad hoc network. Without adequate protection, these packets can be easily subverted or modified. Since the level of trust in a traditional ad hoc network cannot be measured or enforced, enemy nodes or compromised nodes may participate directly in the route discovery, and intercept and filter routing protocol packets to disrupt communication. Compromised users may use the information gleaned from transit packets to mount an attack or anticipate combat moves to their advantage. Since there is no penalty or punishment for misbehavior, in general, nodes have no incentive to behave well. Malicious nodes can insert spurious information into routing packets and cause routing loops, long timeouts, and advertisement of false or exaggerated metrics, replay old routing updates etc.

Traditional ad-hoc routing protocols place complete trust on the nodes and are therefore vulnerable to any or all of these attacks. Other attacks, for example, physically destroying nodes or jamming broadcast signals etc., exist. We recognize that a complete solution to the problem addresses these issues as well. In this paper, we focus on the threats associated with routing protocols and routing misbehavior.

3 Security Aware Ad Hoc Routing (SAR)

We present a general description of our protocol and its behavior and enumerate the metrics we deploy to measure the quality of security of an ad hoc route discovered by our protocol. Originally, ad hoc routing protocols were based on modifications or augmentations to traditional routing protocols for wired networks [4]. These protocols send updates and react to topology changes, using monitoring and other infrastructure support to maintain routing tables. Current research focuses on pure on-demand[8, 13] routing protocols, and more recently, on augmentations that exploit additional information available on the ad-hoc nodes[22, 14, 17] to improve the quality of routes and reduce performance overheads.

Most of the protocols that have been proposed so far focus on discovering the shortest path between two nodes as fast as possible. In other words, the length of the routes is the only metric used in these protocols. Some protocols trade performance and simplified management to obtain bounded sub-optimal paths to speed up the route discovery process[24, 12]. However, the protocol metric is still the length of the routes, measured typically as hop-count. In this paper, we contend that that there are applications that require more than just the assurance that their route has the shortest length. We argue that applications must be able to specify the quality of protection or security attributes of their ad hoc route with respect to metrics that are relevant to them. Our approach shares some similarity with the policy based routing protocols for QoS[5].

3.1 Protocol

For simplicity, we assume that the base protocol is an on-demand protocol similar to AODV or DSR. In the original protocol, when a node wants to communicate with another node, it broadcasts a Route Request or RREQ packet to its neighbors. The RREQ is propagated to neighbors of neighbors and so on, using controlled flooding. The RREQ packets set up a reverse path to the source of the RREQ on intermediate routers that forward this packet. If any intermediate node

has a path already to the RREQ destination, then this intermediate node replies with a Route Reply or RREP packet, using the reverse path to the source. Otherwise, if there exists a route (or connectivity) in the ad hoc network, the RREQ packet will eventually reach the intended destination. The destination node generates a RREP packet, and the reverse path is used to set up a route in the forward direction (RPF or Reverse Path Forwarding).

In SAR, we embed our security metric into the RREQ packet itself, and change the forwarding behavior of the protocol with respect to RREQs. Intermediate nodes receive an RREQ packet with a particular security metric or trust level. SAR ensures that this node can only process the packet or forward it if the node itself can provide the required security or has the required authorization or trust level. If the node cannot provide the required security, the RREQ is dropped. If an end-to-end path with the required security attributes can be found, a suitably modified RREP is sent from an intermediate node or the eventual destination. SAR can be implemented based on any on-demand ad-hoc routing protocol with suitable modification. In this paper, we use AODV[13] as our platform to implement SAR.

3.2 Behavior

Our modification to the traditional ad hoc routing protocol changes the nature of the routes discovered in an ad hoc network. The route discovered by SAR between two communicating entities may not be the shortest route in terms of hop-count. However SAR is able to find a route with a quantifiable guarantee of security. If one or more routes that satisfy the required security attributes exist, SAR will find the shortest such route. If all the nodes on the shortest path (in terms of hop count) between two nodes can satisfy the security requirements, SAR will find routes that are optimal. However, if the ad hoc network does not have a path with nodes that meet RREQ's security requirements, SAR may fail to find a route even if the network is connected.

3.3 Protocol Metrics

In this subsection, we enumerate different techniques to measure or specify the quality of security of a route discovered by our generalized SAR protocol. The first technique is the explicit representation of trust levels using a simple hierarchy that reflects organizational privileges. The next subsection enumerates the different techniques used to protect the integrity of routing messages in fixed-routing protocols.

3.3.1 Trust Hierarchy

SAR provides applications the ability to incorporate explicit trust levels into the route discovery process. Most organizations have an internal hierarchy of privileges. For example, in our battlefield scenario, the military ranks of the users of the ad hoc nodes form an explicit partial-ordering of privilege levels. A simple way of incorporating trust levels into ad hoc networks is to mirror the organizational hierarchy, and associate a number with each privilege level. These numbers represent the security/importance/capability of the mobile nodes and also of the paths. Simple comparison operators can sort these levels to reflect their position in the actual hierarchy. Another alternative is to use what we call the QoP (Quality of Protection) bit vector. For example, if mobile nodes in a network can support four different types of message protection, we can use a four bit vector to represent these message types.

However, what is more important is that this trust level or protection should be immutable. A node with a lower trust level cannot arbitrarily change its trust level, or change the trust level of the

RREQ request it forwards. To provide this guarantee, many techniques can be employed. If keys can be distributed a priori, or a key agreement can be reached by some form of authentication, the simplest technique is to encrypt the portion of the RREQ and RREP headers that contain the trust level. If all the nodes in a trust level share a key, then any node that does not belong to this level cannot decrypt or process the packet, and is forced to drop it. If a node is compromised, tamper-proofing can prevent attackers from learning the values of the keys. In this paper, we leverage related research in key management for ad hoc networks and assume that some mechanism to distribute keys and share secrets is already in place.

3.3.2 Secure Routing Metrics

We develop our notion of the "level of protection" associated with security of information in transit in routing protocol packets. Specifically, in SAR, the aim is to protect any information or behavior that can update or cause a change to the routing tables on cooperating nodes involved in an ad hoc routing protocol. The definition of routing protocol security used here borrows from traditional security services specifications for wired routing protocols [18]. For completeness, timeliness and ordering are added to the list of desirable security properties that can eliminate or reduce the threat of attacks against routing protocols. Techniques that can be used to guarantee these properties are also described. These are shown in Table 1.

Table 1: Secure Ad Hoc Routing - Properties

Property	**Techniques**
Timeliness	Timestamp
Ordering	Sequence Number
Authenticity	Password, Certificate
Authorization	Credential
Integrity	Digest, Digital Signature
Confidentiality	Encryption
Non-repudiation	Chaining of Digital Signatures

The following properties can be integrated into routing protocol messages to prevent attacks that exploit the vulnerability of unprotected information in transit:

- Timeliness: Routing updates need to be delivered in a timely fashion. Update messages that arrive late may not reflect the true state of the links or routers on the network. They can cause incorrect forwarding or even propagate false information and weaken the credibility of the update information. If a node that relays information between two large connected components is advertised as "down" by malicious neighbors, large parts of the network become unreachable. Most ad hoc routing protocols have timestamps and timeout mechanisms to guarantee the freshness of the routes they provide.

- Ordering: Out-of-order updates can also affect the correctness of the routing protocols. These messages may not reflect the true state of the network and may propagate false information. Ad hoc routing protocols have sequence numbers that are unique within the routing domain to keep updates in order.

- Authenticity: Routing updates must originate from authenticated nodes and users. Mutual authentication is the basis of a trust relationship. Simple passwords [10] can be used for weak authentication. Each entity can append a public key certificate, attested by a trusted third party to claim its authenticity. The certifying authority can implement a password based login or a challenge-response mechanism to authenticate the identity in the first place. The receiving node can then verify this claim by examining the certificate. One of the problems in ad hoc networking is the absence of a centralized authority to issue and validate certificates of authenticity.

- Authorization: An authenticated user or node is issued an unforgeable credential by the certificate authority. These credentials specify the privileges and permissions associated by the users or the nodes. Currently, credentials are not used in routing protocol packets, and any packet can trigger update propagations and modifications to the routing table.

- Integrity: The information carried in the routing updates can cause the routing table to change and alter the flow of packets in the network. Therefore, the integrity of the content of these messages must be guaranteed. This can be accomplished by using message digests and digital signatures [16].

- Non-repudiation: Routers cannot repudiate ownership of routing protocol messages they send. A major concern with the updates is the trust model associated with the propagation of updates that originate from distant nodes. Ad-hoc nodes obtain information from their neighbors and forward it to their other neighbors. These neighbors may forward it to other neighbors and so on. In most existing protocols, nodes cannot vouch for the authenticity of updates that are not generated by their immediate neighbors. In order to preserve trust relationships, it becomes necessary to form a chain of routers (using signatures to protect integrity) and authenticate every one in turn, following the chain to the source. This is necessary because trust relationships are not transitive. Alternative solutions that avoid chaining include the path attribute mechanism developed for Secure BGP and secure distance vector routing [18, 2].

- Confidentiality: In addition to integrity, sometimes it may be necessary to prevent intermediate or non-trusted nodes from understanding the contents of packets as they are exchanged between routers. Encrypting the routing protocol packets themselves can prevent unauthorized users from reading it. Only routers that have the decryption key can decrypt these messages and participate in the routing. This is employed when a node cannot trust one or more of its immediate neighbors to route packets correctly, etc.

Each of these desirable properties has a cost and performance penalty associated with it. Some options such as enforcing access control to routing tables using credentials and providing non repudiation by chaining signatures are extremely expensive and impractical to implement and enforce in a generalized routing protocol. However, in scenarios where performance is not the driving factor, a route with quantifiable security guarantees can be more relevant than a shortest route. The purpose of this subsection was to identify the desirable properties of a secure routing protocol. SAR uses security information to dynamically influence the choice of routes installed in the routing tables. Applications can choose to implement a subset of these protection guarantees, based on a cost-benefit analysis of various techniques available to SAR in this decision making phase. In Section 5, we describe a particular implementation of SAR using AODV.

4 Protection

In this section we develop an attack classification and itemize the protection offered by our protocol against attacks on the trust hierarchy and the information in transit in the routing protocol messages. Other attacks on ad hoc networks and related solutions are also briefly discussed.

4.1 Trust levels

Attacks on the trust hierarchy can be broadly classified as Outsider Attacks and Insider Attacks, based on the trust value associated with the **identity** or the source of the attack. SAR modifies the behavior of route discovery, tying in protocol behavior with the trust level of a user. What is also needed is a binding between the identity of the user with the associated trust level. Without this binding, any user can impersonate anybody else and obtain the privileges associated with higher trust levels. To prevent this, stronger access control mechanisms are required (AAA or Authentication, Authorization and Accounting). In order to force the nodes and users to respect the trust hierarchy, cryptographic techniques, e.g., encryption, public key certificates, shared secrets etc., can be employed. For example, all authenticated users belonging to a trust level can share a secret key.

Traditionally strong authentication schemes are used to combat outsider attacks. The identity of a user is certified by a centralized authority, and can be verified using a simple challenge-response protocol. Various schemes including the application of threshold cryptography [25], techniques for key sharing [11], and techniques for key agreement between multiple cooperating entities in dynamic collaborative groups [21] have been proposed to tackle the lack of a centralized authority in an ad hoc network. Our open design allows us to incorporate any of these mechanisms. For example, if one key is used per level, the trust levels are immutable and the trust hierarchy can be enforced. In our implementation, for simplicity, we use a simple shared secret to generate a symmetric encryption/decryption key per trust level. Packets are encrypted using this key and nodes and users belonging to different levels cannot even read the RREQ or RREP packets. Any user or node that is an outsider cannot obtain this key.

Insider attacks are launched by compromised users within a protection domain or trust level. The users may be behaving maliciously, or their identity may be compromised (key is broken etc.). Routing protocol packets in existing ad-hoc algorithms do not carry authenticated identities or authorization credentials, and compromised nodes can potentially cause a lot of damage. Insider attacks are hard to prevent in general at the protocol level. Some techniques to prevent insider attacks include secure transient associations [7], tamper proof or tamper resistant nodes etc. For example, every time a user wants to send a RREQ, the node may require that a user re-key a password, or present her fingerprint for biometric analysis to prove her identity. If the device is lost or captured by an unauthorized user, and an attempt to send RREQs is made, this is detected by the node. The node can then destroy its keys to avoid capture (tamper proofing).

4.2 Information in Transit

In this subsection we examine specific threats to routing protocol **information in transit**. In addition to exploiting vulnerabilities related to the protection and enforcement of the trust levels, compromised or enemy nodes can utilize the information carried in the routing protocol packets to launch attacks. These attacks can lead to corruption of information, disclosure of sensitive information, theft of legitimate service from other protocol entities, or denial of network service to

protocol entities [9]. Threats to information in transit include[20, 9, 19]:

- Interruption: The flow of routing protocol packets, especially route discovery messages and updates can be interrupted or blocked by malicious nodes. Attackers can selectively filter control messages and updates, and force the routing protocol to behave incorrectly. In SAR, a malicious node that interrupts the flow of packets belonging to a higher or lower trust level cannot cause an attack, because it is supposed to drop these packets in any case. If a node filters packets that belong to the same trust level as itself, the broadcast nature of the communication channel can help in detection of interruption attacks by other listeners within transmission range [15].

- Interception and Subversion: Routing protocol traffic and control messages, e.g., the "keep-alive" and "are-you-up?" messages can be deflected, rerouted. In SAR, the messages are protected by the key management infrastructure. In addition, the use of flooding makes these attacks superfluous.

- Modification: The integrity of the information in routing protocol packets can be compromised by modifying the packets themselves. False routes can be propagated, and legitimate nodes can be bypassed. SAR provides a suite of cryptographic techniques that can be incorporated on a need-to-use basis to prevent modification. These include digital signatures and encryption.

- Fabrication: False route and metric information can be inserted into legitimate protocol packets by malicious insider nodes. In such a situation, the sender of the RREQ may receive multiple RREPs. Currently SAR picks the first RREP that arrives at the sender. The sender can be modified to verify that the RREP has credentials that guarantee the integrity of the metrics, and repudiate the ownership of attributes by challenging the intermediate nodes. We plan to incorporate this behavior in the future.

4.3 Other Attacks

Other attacks include replay, timing vulnerabilites and passive attacks. Routing updates that reflect transient topology changes can be stored and retransmitted at a later point of time to trigger false updates and false route propagations. SAR provide automatic replay protection by using sequence numbers and timestamps. Most of the attacks described in this section are also called active attacks as the adversaries actively attempt to change the behavior of the protocol. The complement of these attacks are passive attacks, where the behavior of the adversary is more subtle. Examples of passive attacks include covert channels, traffic analysis, sniffing to compromise keys etc. The information inadvertently disclosed to passive attackers by the protocol packets, can be used to launch active attacks. Protection against eavesdropping or sniffing at the MAC layer can be accomplished by using a suitable MAC layer encryption protocol. Protection against passive attacks are difficult in general and many techniques have been proposed to tackle these problems.

5 Implementation

In this section, we describe an implementation of SAR, built as an augmentation to the AODV[13] protocol in the ns2[1] network simulator. We retain most of ns2 AODV's original behavior, such as on-demand route discovery using flooding, reverse path maintenance in intermediate nodes,

and forward path setup via RREP messages. We modify the RREQ (Route REQuest) and the RREP(Route REPly) packet formats to carry additional security information. We call our modified AODV protocol SAODV (Security-aware AODV).

5.1 SAODV Route Discovery

In SAODV, RREQ packets have an additional field called RQ_SEC_REQUIREMENT that indicates the required security level in the trust hierarchy, for the route the sender wishes to discover. This field is only set once by the sender and does not change during the route discovery phase. When an intermediate node receives a RREQ packet, the protocol first checks if the node can satisfy the security requirement indicated in the packet. If the node is secure enough to participate in the routing, SAODV behaves like AODV and the RREQ packet is forwarded to its neighbors. If the intermediate node cannot satisfy the security requirement, the RREQ packet is dropped and not forwarded. When an intermediate node decides to forward the request, a new field in the RREQ packet is updated. RQ_SEC_GUARANTEE indicates the maximum level of security afforded by the paths discovered. This field is useful in the case where route discovery discovers a route that is more secure than the sender asked for. It is also useful for the security aware applications to get more detailed information about the quality of security for the paths discovered.

This approach opens the question of the effect of malicious nodes in networks. Since it is not uncommon to assume some mobile nodes will either be captured or compromised during the operation[25], SAODV must provide a way to guarantee the cooperation of nodes. This cooperation is achieved by encrypting the RREQ headers, or by adding digital signatures and distributing keys to nodes that belong to the same level in the trust hierarchy that can decrypt these headers and re-encrypt them when necessary.

The arrival of a RREQ packet at the destination indicates the presence of a path from the sender to the receiver that satisfies the security requirement specified by the sender. The destination node sends the RREP packet as in AODV, but with additional information indicating the maximum security available over the path. This information is suitably protected so that only nodes that belong to a particular trust level can process these packets. The value of the RQ_SEC_GUARANTEE field in the RREQ packet is copied to RP_SEC_GUARANTEE field in the RREP packet. When the RREP packet arrives at an intermediate node in the reverse path, intermediate nodes that are allowed to participate, update their routing tables as in AODV and also record the new RP_SEC_GUARANTEE value. This value indicates the maximum security available on the cached forward path. When a trusted intermediate node answers a RREQ query using cached information, this value is compared to the security requirement in the RREQ packet and only when the forward path can guarantee enough security is the cached path information sent back in the RREP.

5.2 Changes to RREQ

SAODV provides support to enforce the trust hierarchy and also enables customizable secure routing. Two new fields are added into the original AODV RREQ packet format to provide support to the trust hierarchy enforcement. The first field is the security requirement, or RQ_SEC_REQUIREMENT. This field is set by the sender and indicates the desired level of trust, within an explicit hierarchy, for the path to the destination. What values to assign to this field is left to the end users. This field can be used to carry simple integer values reflecting the existing hierarchies in a user's organization. For example, if the application is a military situation, the security requirement field can

carry the information about the minimum rank required to relay this communication. In this case, we can use simple integer values to indicate the ranks. A bit vector can also be used to represent combinations of different types of quality of security factors. For example, if the nodes can choose to do simple hash, digital signature, or content encryption over the SAODV routing packets, a three-bit vector can be used to indicate which of those protections the sender wants to have on the new path, in addition to its trust level.

The second field added to the RREQ packet is the security guarantee, or RQ_SEC_GUARANTEE. This field indicates the maximum level of security afforded by all discovered path. It is updated at every hop during the route discovery phase. If RQ_SEC_REQUIREMENT is represented in integers, RQ_SEC_GUARANTEE will be the minimum of the security levels of the participating nodes. If RQ_SEC_REQUIREMENT is represented in bit vectors, RQ_SEC_GUARANTEE will be the result of bit-wise AND operations of all the bit vectors representing the capability of the participating nodes. This information is copied into RREP and sent back to the sender indicating the actual security the sender can use. The sender can use this security guarantee value to determine whether it needs a more secure connection or not. In addition, SAODV also has support for digital signatures. If the application requested integrity support, a new field to store the computed digital signatures was added to the RREQ.

5.3 Changes to RREP

One additional field is also added to the RREP. When an RREQ successfully traverses the network to the sender, the RQ_SEC_GUARANTEE value from RREQ is copied into the RP_SEC_GUARANTEE field inside the RREP packet. The sender can use this value to determine the security level over the whole path. Also, this value is copied into the routing tables of the nodes in the reverse path, to maintain security information about cached paths.

6 Performance Evaluation

This section presents a representative sample of the simulation results collected using our SAODV implementation in ns2 [1]. The original AODV protocol is used as a benchmark to study the pure processing overheads of SAODV. SAODV enables applications to specify security attributes in their routes, and the behavior of SAODV and AODV cannot be compared directly. As our baseline, we use SAODV-D, a protocol that behaves like AODV with respect to its dropping and forwarding behavior, and also includes the additional overheads and modified packet formats of SAODV. The simulation was run for different security attributes, packet formats, traffic patterns, and trust hierarchies. Across our experiments, we observe that SAODV sends fewer routing protocol control messages (RREPs, RREQs, etc.) for the same number of flows and the same amount of application data. As a result, though the overhead per control message is higher in SAODV, the performance impact is sustainable.

6.1 Simulation Set-up

Our results are based on the simulation set up for 50 nodes moving around in 670m by 670m region. Nodes move according to the random way-point model described in [3]. We classify the 50 nodes in our simulations into three levels viz., high, medium and low, each with 15, 15, and 20 nodes respectively. When a node sends out the route request it uses its own security level as the security requirement for the route. In all our measurements, we send the same amount of data (about 10000

packets) and the same number of flows (20), at the same rate. Our simulation is run until the flows complete sending all the packets. Within this set-up, we measure and compare the overall completion time, and the number of control messages sent. Two different traffic patterns are used to drive the simulations. Traffic pattern 1 consists of 20 CBR flows. 10% of the flows are between the high level nodes, 20% between the medium and 70% between the low level nodes. Traffic pattern 2 also has 20 CBR flows, but the distribution is 33%, 33%, 34% for the high, medium, and low level nodes. The packet size is 512 bytes, and the sending rate is 4 packets/second. The maximum number of packets in each flow is 500. In subsection 6.2 we measure the overheads for enforcing the trust hierarchy. Section 6.3 presents our results for secure routing, specifically, the overheads of adding encryption and digital signatures to SAODV's RREPs.

6.2 SAODV Processing Overheads

SAODV has larger RREQ and RREP packets compared to AODV and all the nodes participating in the route discovery must do additional processing to check the security requirement and update the security guarantee. Initially, SAODV is configured to do the trust enforcement processing, but not drop the RREQ packets when it is supposed to. As mentioned earlier, this version is a baseline in performance measurement to show how SAODV processing affects protocol behavior. The pure overhead of the SAODV modification is measured in comparison to the original AODV.

Table 2: Overall Simulation Time

	Traffic Pattern 1	**Traffic Pattern 2**
AODV	2803	2844
SAODV-D	2844	2918

In Table 2 With the traffic pattern 1, SAODV takes 1% longer time to finish the whole simulation and with the traffic pattern 2, less than 3% more. This means that the pure overhead of adding additional processing to enable security, in the absence of dropping, is not prohibiting. We use this SAODV without RREQ dropping, SAODV-D, as our baseline for rest of the performance measurements.

6.2.1 Path Discovery

Next, we ran SAODV-D and SAODV with explicit trust values, on the same traffic patterns to observe the difference in protocol behavior. The number of paths discovered by SAODV-D and SAODV, and the number of paths that violate the security requirement in SAODV-D were recorded. Since SAODV-D behaves like original AODV, some of the paths found violated the security requirement. This is summarized in Table 3.

Table 3: Number of Paths Discovered

	Traffic Pattern 1	**Traffic Pattern 2**
Total number of path discovery by SAODV-D	93	95
Paths violating security requirement by SAODV-D	14	19
Paths discovery by SAODV	80	73

Though SAODV-D found more paths when the trust levels were enforced, 14 and 19 of these paths respectively were unusable. SAODV discovered fewer paths, but these paths are guaranteed to obey the trust requirements of their senders.

6.2.2 Routing Message Overheads

Table 4 shows the numbers of routing protocol messages in SAODV-D and SAODV. We observe that there is a drop in the number of RREQ messages in sent in SAODV. This is because the RREQ is dropped and not forwarded when the intermediate nodes cannot handle the security requirement of the RREQ packets.

Table 4: Routing Message Overhead

	RREQ		RREP		Routing Msgs	
	Pattern 1	Pattern 2	Pattern 1	Pattern 2	Pattern 1	Pattern 2
SAODV-D	2333	2566	107	102	2410	2668
SAODV	2285	1504	80	73	2365	1577

These numbers imply that SAODV generates fewer routing messages, while enabling applications to find more relevant routes. In the case of Pattern 1, there was a decrease of 2% in RREQ messages and 25% in RREP messages. For Pattern 2, the results were more accentuated (41% in RREQs, and 27% in RREPs). This is due to the fact that the trust hierarchy is more equitably distributed in Pattern 2 and paths tend to be smaller.

6.2.3 Overall Simulation Time and Transmitted Data

SAODV restricts the route discovery process only to paths that can satisfy the requirements. This feature may force packets to follow longer, but more secure paths and result in taking more time to finish the communication. The overhead of the protocol is illustrated in Table 5. The overall time to complete transmission of all the traffic flows in both SAODV with trust enforcement and SAODV-D, and the total amount of data transmitted are illustrated in the Table.

Table 5: Overall Simulation Time and Transmitted Data

	Simulation Time		Transmitted Data	
	Pattern 1	Pattern 2	Pattern 1	Pattern 2
SAODV-D	2844	2918	10023	10022
SAODV	2911	2925	10028	10017

With RREQ dropping, SAODV takes 2.3% and 0.2% more time to finish in traffic patterns 1 and 2 compared to SAODV-D. Although SAODV takes marginally more time to finish communication, it still finds paths in most cases, and delivers almost the same amount of data from senders to the receivers as shown in the Table.

6.2.4 Path Optimality

The data packets may follow longer paths in SAODV when the shorter routes cannot satisfy the security requirements. Table 6 shows the length of the paths each data packet travels compared to

the shortest possible paths at the time. The optimal length means that the packet actually follows the shortest path between the sender and the destination at the time of transmission. The table lists the the number of packets traversed the path with each length. With Pattern 1, we see that a significant number of packets take longer paths, especially among transmissions in the higher trust levels (34%). The impact is not as severe in traffic pattern 2 (14%).

Table 6: Path Optimality

	Optimal		Optimal + 1		Optimal + 2		Optimal + 3	
	Pat. 1	Pat. 2	Pat. 1	Pat. 2	Pat. 1	Pat. 2	Pat. 1	Pat. 2
SAODV-D	7700	8570	2142	1266	130	132	0	4
SAODV	6481	8549	2414	1273	1075	130	0	3

6.3 Secure Routing Measurements

The overhead of including security attributes in the RREQ messages are presented in this subsection. The SAODV protocol is augmented with hash digests and symmetric encryption mechanisms. The signed hash digests provide message integrity, whereas encrypting packets guarantees their confidentiality. Nodes that have the same trust level share the same encryption and decryption keys. The MD5 Hash algorithm and the Blowfish block cipher were used for these measurements. We present the measurements for Traffic Pattern 1 only, due to space constraints. The results for Pattern 2 show a similar trend.

Table 7: Routing Message Overheads for Secure Routing

	RREQ		RREP		Routing Msgs	
	Encryption	Signed Hash	Encryption	Signed Hash	Encryption	Signed Hash
SAODV-D	2225	2219	77	85	2378	2381
SAODV	2175	2148	74	80	2341	2311

The entire RREQ packet was encrypted, with the exception of the packet-type field. For SAODV with digital signatures, an additional field was added to the RREQ header. The MD5 has algorithm was used to generate a MAC (Message Authentication Code), along with Blowfish encryption to protect the integrity of the MAC. The SAODV-D protocol reflects the overhead of adding the extra field in the header. In Table 7, we observe that SAODV/Encryption and SAODV/Digital Signature sent fewer RREQs and RREPs than SAODV-D. This is because nodes that were not capable of decrypting the encrypted RREQ packets, or couldn't verify the signatures, dropped these packets without forwarding. SAODV with Encryption showed a 9.1% decrease and SAODV/Digital Signatures showed a 17% decrease. This reinforces our claim that SAODV sends fewer control messages (RREQs and RREPs) than SAODV-D, though each packet needs more processing.

Table 8 presents overall simulation time and transmitted data. Adding Encryption increases overall simulation time by 0.7% , and adding digital signatures by 2% (in addition to the trust enforcement overheads). However, the number of packets transmitted was approximately equal.

Table 8: Overall Simulation Time and Transmitted Data

	Simulation Time		Transmitted Data	
	Encryption	Signed Hash	Encryption	Signed Hash
SAODV-D	2899	2875	10024	10025
SAODV	2918	2933	10026	10017

7 Conclusion

SAR enables the discovery of secure routes in a mobile ad hoc environment. Its integrated security metrics allow applications to explicitly capture and enforce explicit cooperative trust relationships. In addition, SAR also provides customizable security (e.g., encryption for confidentiality etc.) to the flow of routing protocol messages themselves. Routes discovered by SAR come with "quality of protection" guarantees. The techniques enabled by SAR can be easily incorporated into generic ad hoc routing protocols as illustrated by our implementation example - SAODV. The processing overheads in SAR are offset by restricting the scope of the flooding for more relevant routes, providing comparable price/performance benefits.

References

[1] The Network Simulator - NS-2. http://www.isi.edu/nsnam/ns/.

[2] B. Smith and S. Murthy and J.J. Garcia-Luna-Aceves. Securing Distance Vector Routing Protocols. In *Internet Society Symposium on Network and Distributed System Security, the 7th International Workshop on Security Protocols*, San Diego, CA, USA, February 1997.

[3] J. Broch, D. A. Maltz, D. B. Johnson, Yih-Chun Hu, and J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, TX, USA, October 1998.

[4] C. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, 1994.

[5] E. Crawley and R. Nair and B. Rajagopalanand and H. Sandick. A Framework for QoS-based Routing in the Internet. RFC 2386, August 1998.

[6] E. M. Royer and C-K Toh. A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks. *IEEE Personal Communications*, April 1999.

[7] F. Stajano and R. Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *The 7th International Workshop on Security Protocols*, Cambridge, UK, April 1999.

[8] J. Broch and D. B. Johnson. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. IETF Internet Draft, October 1999.

[9] J. Howard. *An Analysis Of Security Incidents On The Internet 1989 - 1995.* PhD thesis, Doctor of Philosophy in Engineering and Public Policy, Carnegie Mellon University, April 1997.

[10] J. Moy. OSPF Version 2. RFC 2326, April 1998.

[11] N. Asokan and P. Ginzboorg. Key-Agreement in Ad-hoc Networks. In *The Fourth Nordic Workshop on Secure Computer Systems*, 1999.

[12] P. Sinha and R. Sivakumar and V. Bharghavan. CEDAR: a Core-Extraction Distributed Ad Hoc Routing algorithm. In *The 18th Annual Joint Conference of the IEEE Computer and Communication Societies*, New York, NY, USA, March 1999.

[13] C. E. Perkins and E. M. Royer. Ad-hoc On-Demand Distance Vector Routing. In *The Second IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, USA, February 1999.

[14] R. Dube and C. D. Rais and Kuang-Yeh Wang and S. K. Tripathi. Signal stability-based adaptive routing (SSA) for ad hoc mobile networks. *IEEE Personal Communications*, February 1997.

[15] S. Marti and T. Giuli and K. Lai and M. Baker. Mitigating Routing Misbehavior in Mobile ad hoc networks. In *The Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Boston, MA, USA, August 2000.

[16] S. Murphy and M. Badger and B. Wellington. OSPF with Digital Signatures. RFC 2154.

[17] S. Singh and M. Woo and C. S. Raghavendra. Power-Aware Routing in Mobile Ad Hoc Networks. In *The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, TX, USA, October 1998.

[18] B. Smith, S. Murthy, and J.J. Garcia-Luna-Aceves. Securing the Border Gateway Routing Protocols. In *Global Internet '96*, London, UK, November 1996.

[19] W. Stallings. *Network and Internetwork Security Principles and Practice.* Prentice Hall, Englewood Cliffs, NJ, 1995.

[20] F. Wang, Brian Vetter, and Shyhtsun Felix Wu. Secure routing protocols: Theory and practice. Technical Report, North Carolina State University.

[21] Y. Kim and A. Perrig and G. Tsudik. Simple and fault-tolerant key agreement for dynamic collaborative groups. In *ACM Conference on Computer and Communications Security*, pages 235–244, 2000.

[22] Y. Ko and N. H. Vaidya. Location-Aided Routing(LAR) in Mobile Ad Hoc Networks. In *The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, TX, USA, October 1998.

[23] Y. Zhang and W. Lee. Intrusion Detection in Wireless Ad-Hoc Networks. In *The Sixth Annual ACM/IEEE Conference on Mobile Computing and Networking*, Boston, MA, USA, August 2000.

[24] Z. Haas and M. Pearlman. The zone routing protocol (ZRP) for ad hoc networks. Internet draft, draft-zone-routing-protocol-00.txt, 1997.

[25] L. Zhou and Z. J. Haas. Securing Ad Hoc Networks. *IEEE Network Magazine*, November 1999.