A REVIEW PAPER ON AD HOC NETWORK SECURITY

Karan Singh, karancs12@yahoo.com
Computer Science and Engineering Department
Motilal National Institute of Technology, Allahabad
Allahabad, India -211004

Rama Shankar Yadav, rsy@mnnit.ac.in
Computer Science and Engineering Department
Motilal National Institute of Technology, Allahabad
Allahabad, India -211004

Ranvijay, cs0620@mnnit.ac.in
Computer Science and Engineering Department
Motilal National Institute of Technology, Allahabad
Allahabad, India -211004

Abstract

In this article we present a survey of secure ad hoc routing protocols for wireless networks. Ad hoc network is a collection of nodes that is connected through a wireless medium forming rapidly changing topologies. Attacks on ad hoc network routing protocols disrupt network performance and reliability with there solution. We briefly present the most popular protocols that follow the table-driven and the source-initiated on-demand approaches. The comparison between the proposed solutions and parameters of ad hoc network shows the performance according to secure protocols. We discuss in this paper routing protocol and challenges and also discuss authentication in ad hoc network.

KEYWORDS: Wireless Network, Ad hoc Network, Security Service, Routing Protocols, Routing Authentication, Hash function and Secure Routing Protocols.

I. INTRODUCTION

Wireless networks [34] consist of a number of nodes which communicate with each other over a wireless channel which have various types of networks: sensor network, ad hoc mobile networks, cellular networks and satellite networks. Wireless sensor networks consist of small nodes with sensing, computation and wireless communications capabilities. Many routing protocols have been specifically designed for WSNs where energy awareness is the key issue. Routing protocols in WSNs [41] differ depending on the application and network architecture. Ad-hoc networks are a new paradigm of wireless communication for mobile hosts where node mobility causes frequent changes in topology. Ad hoc networks are self-configurable and autonomous systems consisting of routers and hosts, which are able to support movablity and organize themselves arbitrarily. This means that the topology of the ad hoc network changes dynamically and unpredictably. Moreover, the ad hoc network can be either constructed or destructed quickly and autonomously without any administrative server or infrastructure. Without support from the fixed infrastructure, it is undoubtedly arduous for people to distinguish the insider and outsider of the wireless network. That is to say, it is not easy for us to tell apart the legal and the illegal participants in wireless systems. Because of the above mentioned properties, the implementation of security infrastructure has become a critical challenge when we design a wireless network system. If the

nodes of ad hoc networks are mobile and with wireless communication to maintain the connectivity, it is known as mobile ad hoc network (MANET) and require an extremely flexible technology for establishing communications in situations which demand a fully decentralized network without any fixed base stations, such as battlefields, military applications, and other emergency and disaster situations.

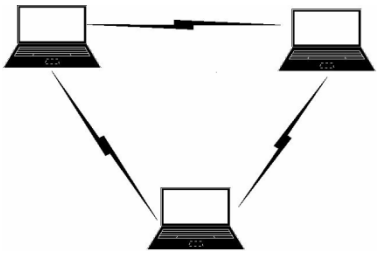

FIGURE 1: AD HOC NETWORK

Since, all nodes are mobile, the network topology of a MANET is generally dynamic and may change frequently. Thus, protocol such as 802.11 to communicate via same frequency or Bluetooth have require power consumption is directly proportional to the distance between hosts, direct *single-hop* transmissions between two hosts can require significant power, causing interference with other such transmissions [41]. To avoid this *routing problem*, two hosts can use *multi-hop* [34] transmission to communicate via other hosts in the network A router should provide the ability to rank routing information sources from most trustworthy to least trustworthy and to accept routing information about any particular destination from the most trustworthy sources first. A router should provide a mechanism to filter out obviously invalid routes. Routers must not by default redistributes routing data they do not themselves use, trust or otherwise consider valid. Routers must be at least a little paranoid about accepting routing data from anyone, and must be especially careful when they distribute routing information provided to them by another party.

Figure 1 shows three node where ad hoc network where every node is connected to wireless, and work as access point to forward and receive data. This article discuss attacks on ad hoc networks and discusses current approaches for establishing cryptographic keys in ad hoc networks. We describe the state of research in secure ad hoc routing protocols, routing challenges and its research issues.

II. ROUTING PROTOCOL AND ITS CHALLENGE IN AD HOC NETWORK

In this section we are going to discuss different approaches adopted for routing and security challenges in Ad hoc networks.

A. ROUTING PROTOCOLS

Routing in mobile ad hoc networks faces additional problems and challenges [22], [30] when compared to routing in traditional wired networks with fixed infrastructure. There are several well known protocols in the literature that have been specifically developed to cope with the limitations imposed by ad hoc networking environments. Most of the existing routing protocols follow two different design approaches to confront the inherent Characteristics of ad hoc networks: the *table-driven* and the *source-initiated on-demand* approaches.

Table-driven ad hoc routing protocols maintain at all times routing information regarding the connectivity of every node to all other nodes that participate in the network. Also known as *proactive*, [49] these protocols allow every node to have a clear and consistent view of the network topology by propagating periodic updates [27]. An alternative approach to that followed by table-driven protocols is the source-initiated on-demand routing. According to this approach, a route is created only when the source node requires a route to a specific destination. A route is acquired by the initiation of a *route discovery* function by the source node.

The data packets transmitted while a route discovery is in process are buffered and are sent when the path is established. An established route is maintained as long as it is required through a *route maintenance* procedure. Table 1 shows the various type of routing protocols according to parameter which are response time, bandwidth and energy.

Parameter	Network	Protocols	Examples
Response Time And Bandwidth	Ad hoc	Proactive protocols	Destination-sequenced Distance-Vector (DSDV)
			Optimized Link- State Routing (OLSR)
		Reactive protocols	Ad Hoc On-Demand Distance-Vector (AODV)
			Dynamic Source Routing (DSR)
			Geography-based routing
			Cluster-based (or *hierarchical*) routing
Energy	Sensor	Network structure	Flat network routing
			Hierarchical network routing
			Location based routing
		Protocol operation	Negotiation based routing
			Multi-path based routing
			Query based routing
			QoS based routing
			Coherent based routing

TABLE 1: CLASSIFICATION OF ROUTING PROTOCAL

B. SECURITY CHALLENGES IN AD HOC NETWORKS

Use of wireless links renders an Ad hoc network susceptible to link attacks ranging from passive eavesdropping to active impersonation, message replay and message distortion [9],[10],[52].Eavesdropping might give an attacker access to secret information thus violating confidentiality. Active attacks could range from deleting messages, injecting erroneous messages; impersonate a node etc thus violating availability, integrity, authentication and non-repudiation. Nodes roaming freely in a hostile environment with relatively poor physical protection have non-negligible probability of being compromised. Hence, we need to consider malicious attacks not only from outside but also from within the network from compromised nodes. Thus following are the ways by which security can be breached. [56]

- **Vulnerability of Channels:** As in any wireless network, messages can be eavesdropped and fake messages can be injected into the network without the difficulty of having physical access to network components.
- **Vulnerability of nodes:** Since the network nodes usually do not reside in physically protected places, such as locked rooms, they can more easily be captured and fall under the control of an attacker.
- **Absence of Infrastructure:** Ad hoc networks are supposed to operate independently of any fixed infrastructure. This makes the classical security solutions based on certification authorities and on-line servers inapplicable.
- **Dynamically Changing Topology**: In mobile ad hoc networks, the permanent changes of topology require sophisticated routing protocols, the security of which is an additional challenge. A particular difficulty is that incorrect routing information can be generated by compromised nodes or as a result of some topology changes and it is hard to distinguish between the two cases.

For high survivability Ad hoc networks should have a distributed architecture with no central entities, centrality increases vulnerability. Ad-hoc network is dynamic due to frequent changes in topology. Even the trust relationships among individual nodes also changes, especially when

some nodes are found to be compromised. Security mechanism need to be on the dynamic and not static and should be scalable.

III. SECURITY MODEL

In this section we first discuss security goals attacks and thus secure routing protocol which are following:

A. SECURITY GOALS FOR AD HOC

- **Availability:** Ensures survivability despite Denial Of Service (DOS) attacks. On physical and media access control layer attacker can use jamming techniques to interfere with communication on physical channel. On network layer the attacker can disrupt the routing protocol. On higher layers, the attacker could bring down high level services e.g.: key management service.
- **Confidentiality:** Ensures certain information is never disclosed to unauthorized entities.
- **Integrity:** Message being transmitted is never corrupted.
- **Authentication:** Enables a node to ensure the identity of the peer node it is communicating with. Without which an attacker would impersonate a node, thus gaining unauthorized access to resource and sensitive information and interfering with operation of other nodes.
- **Non-repudiation:** Ensures that the origin of a message cannot deny having sent the message.
- **Non-impersonation**: No one else can pretend to be another authorized member to learn any useful information.
- **Attacks using fabrication:** Generation of false routing messages is termed as fabrication messages. Such attacks are difficult to detect.

B. ATTACK ON AD HOC NETWORK

There are various types of attacks on ad hoc network which are describing following:

- **Location Disclosure:** Location disclosure is an attack that targets the privacy requirements of an ad hoc network. Through the use of traffic analysis techniques [20], or with simpler probing and monitoring approaches, an attacker is able to discover the location of a node, or even the structure of the entire network.
- **Black Hole:** In a black hole attack a malicious node injects false route replies to the route requests it receives, advertising itself as having the shortest path to a destination[26]. These fake replies can be fabricated to divert network traffic through the malicious node for eavesdropping, or simply to attract all traffic to it in order to perform a denial of service attack by dropping the received packets.
- **Replay:** An attacker that performs a replay attack injects into the network routing traffic that has been captured previously. This attack usually targets the freshness of routes, but can also be used to undermine poorly designed security solutions.
- **Wormhole:** The wormhole attack is one of the most powerful presented here since it involves the cooperation between two malicious nodes that participate in the network [53]. One attacker, e.g. node A, captures routing traffic at one point of the network and tunnels them to another point in the network, to node B, for example, that shares a private communication link with A. Node B then selectively injects tunneled traffic back into the network. The connectivity of the nodes that have established routes over the wormhole link is completely under the control of the two colluding attackers. The solution to the wormhole attack is *packet leashes*.
- **Blackmail:** This attack is relevant against routing protocols that use mechanisms for the identification of malicious nodes and propagate messages that try to blacklist the offender [58]. An attacker may fabricate such reporting messages and try to isolate legitimate nodes from the network. The security property of non-repudiation can prove to be useful in such cases since it binds a node to the messages it generated.

- **Denial of Service:** Denial of service attacks aim at the complete disruption of the routing function and therefore the entire operation of the ad hoc network [15]. Specific instances of denial of service attacks include the *routing table overflow and* the *sleep deprivation torture.*. In a routing table overflow attack the malicious node floods the network with bogus route creation packets in order to consume the resources of the participating nodes and disrupt the establishment of legitimate routes. The sleep deprivation torture attack aims at the consumption of batteries of a specific node by constantly keeping it engaged in routing decisions.
- **Routing Table Poisoning:** Routing protocols maintain tables that hold information regarding routes of the network. In poisoning attacks the malicious nodes generate and send fabricated signaling traffic, or modify legitimate messages from other nodes, in order to create false entries in the tables of the participating nodes [15]. For example, an attacker can send routing updates that do not correspond to actual changes in the topology of the ad hoc network. Routing table poisoning attacks can result in the selection of non-optimal routes, the creation of routing loops, bottlenecks, and even portioning certain parts of the network.
- **Rushing Attack:** Rushing attack is that results in denial-of-service when used against *all* previous on-demand ad hoc network routing protocols [55]. For example, DSR, AODV, and secure protocols based on them, such as Ariadne, ARAN, and SAODV, are unable to discover routes longer than two hops when subject to this attack. develop *Rushing Attack Prevention (RAP)*, a generic defense against the rushing attack for on-demand protocols that can be applied to any existing on-demand routing protocol to allow that protocol to resist the rushing attack.
- **Breaking the neighbor relationship:** An intelligent filter is placed by an intruder on a communication link between two ISs(Information system) could modify or change information in the routing updates or even intercept traffic belonging to any data session.
- **Masquerading:** During the neighbor acquisition process, a outside intruder could masquerade an nonexistent or existing IS by attaching itself to communication link and illegally joining in the routing protocol do main by compromising authentication system. The threat of masquerading is almost the same as that of a compromised IS.
- **Passive Listening and traffic analysis:** The intruder could passively gather exposed routing information. Such a attack can not effect the operation of routing protocol, but it is a breach of user trust to routing the protocol. Thus, sensitive routing information should be protected. However, the confidentiality of user data is not the responsibility of routing protocol

C. ROUTING SECURITY IN AD HOC NETWORK

The contemporary routing protocols for Ad hoc networks cope well with dynamically changing topology but are not designed to accommodate defense against malicious attackers. No single standard protocols capture common security threats and provide guidelines to secure routing. Routers exchange network topology informally in order to establish routes between nodes another potential target for malicious attackers who intend to bring down the network. External attackers injecting erroneous routing info, replaying old routing info or distorting routing info in order to partition a network or overloading a network with retransmissions and inefficient routing. Internal compromised nodes - more severe detection and correction more difficult Routing info signed by each node won't work since compromised nodes can generate valid signatures using their private keys.

Detection of compromised nodes through routing information is also difficult due to dynamic topology of Ad hoc networks [22]. Routing protocols for Ad hoc networks must handle outdated routing information to accommodate dynamic changing topology. False routing information generated by compromised nodes can also be regarded as outdated routing information. As long as there are sufficient numbers of valid nodes, the routing protocol should be able to bypass the compromised nodes, this however needs the existence of multiple, possibly disjoint routes between nodes. Routing protocol should be able to make use of an alternate route if the existing one appears to have faulted

D. ROUTING AUTHENTICATION

Routing authentication is one of the important factors in ad hoc networks during route discovery because ad hoc is infrastructure less network. So it is required that a reply coming from a node against a route request must be authentic. That's why authentication protocol is required between the nodes of ad hoc network. In this section we emphasize on the ways by which these protocols can be used.

i. New key agreement scenario

Consider a group of people getting together for an Ad hoc meeting in a room and trying to establish a wireless network through their laptops. They trust one another personally; however don't have any a priori shared secret (password) to authenticate one another. They don't want anybody outside the room to get a wind of their conversation indoors. This particular scenario is vulnerable to any attacker who not only can monitor the communication but can also modify the messages and can also insert messages and make them appear to have come from somebody inside the room. This is a classic example of Ad hoc network and the simplest way to tackle this example would be through location based key agreement - to map locations to name and then use identity based mechanisms for key agreement[10][56]. e.g.: participants writing the IP addresses on a piece of paper and passing it around. Then a certificate based key agreement mechanism can be used. These public key certificates can allow participants to verify the binding between the IP address and keys of other participants.

ii. Two obvious problems

a) Difficult to determine if the certificate presented by the participant has been revoked
b) Participants may be divided into 2 or more certification hierarchies and that they don't have cross certification hierarchies.

One obvious solution

A trusted third is party capable of locating players, however not always feasible due to non-infrastructure nature of Ad hoc networks. Physically secure channel limited to those present in the room to negotiate the session key before switching to the insecure wireless channel.

iii. Password based Authenticated Key Exchange

A fresh password is chosen and shared among those present in the room in order to capture the existing shared context. If this password is long random string, can be used to setup security association, but less user friendly. Natural language phrases are more users friendly, however vulnerable to dictionary attacks[10][16][42]. Need to derive a strong session key from a weak shared password. Desirable properties for such a protocol are following

- **Secrecy:** Only those players that know the initial shared weak secret password should learn the session key and nobody else should.
- **Perfect Forward Secrecy:** Warrants that if an attacker who succeeds in compromising one of the participants at a later time would be unable to figure out the session key resulting from previous runs of protocol.
- **Contributory Key Agreement:** If each and every player participates in the creation of the final session key, by making a contribution, then it is called contributory key agreement.
- **Tolerance to Disruption Attempts:** Not only strong attackers who can disrupt communication by jamming radio channels etc but even the weaker attackers who can insert but cannot modify or delete messages sent by players are also provided for.

iv. Password authenticated Diffie - Hellman key Exchange

- **Two Party Version:** In the elementary DH protocol, *two parties* A and B agree on a prime p and a generator g of the multiplicative group Zp^* (i.e. the set $\{1, 2 \dots p-1\}$). A and B choose random secrets S_A and S_B such that $1 \leq S_A, S_B \leq p-1$
  1) A computes g^{S_A}, encrypts it with the shared secret password P and sends it to B. $A \rightarrow B: A,\ P(g^{S_A})$

2) B extracts g^{S_A} from the message computes g^{S_B} and also computes the session key $K = (g^{S_A})^{S_B}$. B then chooses a random challenge C_B and encrypts it using the key K. B encrypts S_B using P. It then sends the two quantities to A. $B \rightarrow A: P(S_B), K(C_B)$.
3) A extracts S_B from $P(S_B)$ and computes the key $K = (g^{S_A})^{S_B}$. It then extracts C_B by decrypting $K(C_B)$. A then generates challenge (random) C_A, encrypts both C_A and C_B with K and sends it to B. $A \rightarrow B: K(C_A, C_B)$.
4) This message (3) convinces B that A was able to decrypt the message in (2) correctly. B then encrypts C_A using K and sends it to A. $B \rightarrow A: K(C_A)$.

A decrypts the message to see if the plaintext is indeed C_A. This would convince A that B knew K. This would in turn convince A that B knew P.

- **Multi-party version:** There are let's just say n players M_1, M_2, …, M_n who all share a password P, each generating a random quantity S_i which is its contribution to the eventual session key K = $g^{S1S2\_\_\_Sn\text{-}1Sn}$. The protocol is divided into 3 parts. In the first part (steps 1 and 2) players M_i to M_{n-1} generate an intermediate key $PI = g^{S1S2\text{---}Sn-1}$ *in* $n-1$ steps.

In the second part (steps 3 and 4) each M_i (where i = 1 to n-1) has a separate with M_n, at the end of which all the players are in a position to compute K. The third part (step 5) being the key confirmation.

1) $M_i \rightarrow M_{i+1}: PI = g^{S1S2\text{---}Si}$ $i = 1$ *to* $n-2$ in sequence.
2) M_{n-1} --> ALL : PI = $g^{S1S2\_\_\_Sn\text{-}1}$, broadcast
3) $M_i \rightarrow M_n: P(C_i)$ $i = 1$ *to* $n-1$, in parallel, where C_i = PI $^{Si'/Si}$ and S_i' is the blinding factor that is randomly chosen by M_i.
4) $M_n \rightarrow M_i: (C_i)S_n$ $i = 1$ *to* $n-1$, in parallel.
5) $M_i \rightarrow ALL: K(M_i, h(M_1, M_2, \ldots\ldots, M_n))$ broadcast.

Step 1 consists of (n-2) sub steps. In the first sub step player M_1 computes g^{S1} and sends it to M_2 etc. At the end of the (n-2)th sub step, M_{n-1} receives $g^{S1S2\_\_\_Sn\text{-}2}$, which it then raises by (S_{n-1}) to get the intermediate key PI = $g^{S1S2\text{---}Sn\text{-}1}$.

In step 2, M_{n-1} broadcast this PI to everyone. Now every M_i *(i = 1 to n-1)* removes its contribution i.e, S_i *(i=1 to n-1)* from the PI respectively but also inserts a randomly chosen blinding factor S_i, encrypts the whole thing with the shared password P.

In step 3, each M_i will in parallel send the encryption to M_n. M_n decrypts the received message to extract C_i. It then raises each C_i by S_n and returns the result in parallel to each M_i. At this point each player can compute the session key as follows K = $g^{S1S2\_\_\_Sn\text{-}1Sn}$. M_n raises PI by S_n *: K = (PI)*^{Sn}. Each M_i unblinds the quantity it receives from M_n and re inserts its original contribution S_i to construct the session key K = $g^{S1S2\_\_\_Sn\text{-}1Sn}$ = $(PI)^{Sn}$.

Finally, some player broadcasts a key confirmation message that allows each player to verify that at least one another player has decided on the same key K. The blinding factor S_i is needed for the following reasons.

1) Without the blinding, the quantity encrypted with P by M_{n-1} from step 3 is the same as what it receives in step 1.
2) An attacker could send $g^{S1S2\_\_\_Si}$ to M_i in step 2 instead of the broadcast message (intermediate key) PI. If M_i uses this quantity to generate its message in step 3, the resulting message is same as the message received by M_i in step 1. To thwart dictionary attacks, blinding is necessary.

This protocol does provide perfect forward secrecy. It is also quasi-resilient to disruption except when M_n is compromised.

IV. SECURE ROUTING PROTOCOLS

A. ARAN

Authenticated Routing for Ad-hoc Networks (ARAN) detects and protects against malicious actions by third parties and peers in Ad-hoc environment. ARAN introduces authentication, message integrity and non-repudiation to an Ad-hoc environment [12][30]. ARAN is composed of two distinct stages. The first stage is simple and requires little extra work from peers beyond traditional ad hoc protocols. Nodes that perform the optional second stage increase the security of their route, but incur additional cost for their ad hoc peers who may not comply (e.g., if they are low on battery resources). Brief description of stages as follows

- **Stage 1:** It contains a preliminary certification stage and a mandatory end-end authentication stage. It is a lightweight stage and does not demand too many resources.
  Preliminary Certification: ARAN requires the use of a trusted certificate server T. Before entering the Ad-hoc network, each node requests a certificate from T. For a node A the certificate contains the IP address of A, the public key of A, a timestamp t of when the certificate was created, and a time e at which the certificate expires. These variables are concatenated and signed by T. All nodes must maintain fresh certificates with the trusted server and must know T's public key.
  End-to-End Authentication: The goal of stage 1 is for the source to verify that the intended destination was reached. In this stage, the source trusts the destination to choose the return path.
  Source node: A source node A, begins route instantiation to a destination X by broadcasting to its neighbors a route discovery packet (RDP): $\rightarrow broadcast: [RDP, IP_X, Cert_A, N_A, t]K_A -$. The RDP includes a packet type identifier ("RDP"), the IP address of t the destination (IPx), A's certificate (CertA), a nonce N_A , and the current time t, all signed with A's private key. Each time A performs route discovery, monotonically increases the nonce. Nodes then store the nonce they have last seen with its timestamp.
  Intermediate node for RDP: Each node records the neighbor from which it received the message. It then forwards the message to each of I its neighbors, signing the contents of the message. This signature prevents poofing attacks that may alter the route or form loops. Let A's neighbor be B. $B \rightarrow broadcast:$ $[[RDP, IP_X, Cert_A, N_A, t]K_A -]K_B -, Cert_B$. Nodes do not forward messages for which they have already seen the (N_A ,IP_A) tuple. Upon receiving the broadcast, B's neighbor C validates the signature with the given certificate. C then rebroadcasts the RDP to its neighbors, first removing B's signature. $C \rightarrow broadcast: [[RDP, IP_X, Cert_A, N_A, t]K_A -]K_C -, Cert_C$
  Destination node: Eventually, the message is received by the destination, X, who replies to the first RDP that it receives for a source and a given nonce. There is no guarantee that the first RDP received traveled along the shortest path from the source. The destination unicasts a Reply (REP) packet back along the reverse path to the source. $X \rightarrow D: [REP, IP_A, Cert_X, N_A, t]K_X -$
  Intermediate node for REP : Nodes that receive the REP forward the packet back to the predecessor from which they received the original RDP. All REPs are signed by the sender. Let D's next hop to the source be node C. $D \rightarrow C: [[REP, IP_A, Cert_X, N_A, t]K_X -]K_D -, Cert_D$ C validates D's signature, removes the signature, and then signs the contents of the message before unicasting the RDP to B. $C \rightarrow B: [[REP, IP_A, Cert_X, N_A, t]K_X -]K_C -, Cert_C$ A node checks the signature of the previous hop as the REP is returned to the source. This avoids attacks where malicious nodes instantiate routes by impersonation and re-play of X's message.
  Source Node: When the source receives the REP, it verifies that the correct nonce was returned by the destination as well as the destination's signature. Only the destination can answer an RDP packet. Other nodes that already have paths to the destination cannot reply for the destination. While other protocols allow this networking optimization, we note that removing it also removes several possible exploits and cuts down on the reply traffic received by the source. Because only the destination can send REPs, loop freedom is guaranteed easily.

Disadvantages: ARAN requires that nodes keep one routing table entry per source-destination pair that is currently active. This is certainly more costly than per-destination entries in non-secure ad hoc routing protocols.

- **Stage 2:** It is done only after Stage 1 is over. This is because the destination certificate is required in Stage 2. This stage is primarily used for discovery of shortest path in a secure fashion. Since a path is already discovered in Stage 1, data transfer can be pipelined with Stage 2 's shortest path discovery operation.
  Source Node: The source begins by broadcasting a Shortest Path Confirmation (SPC) message to its neighbors (the same variables are used as in stage 1. $A \rightarrow broadcast: SPC, IP_X, Cert_X, [[IP_X, Cert_A, N_A, t]K_A -]K_X +$. The SPC message begins with the SPC packet identifier ("SPC"), X's IP address and certificate. The source concatenates a signed message containing the IP address of X, its certificate, a nonce and timestamp. This signed message is encrypted with X's public key so that other nodes cannot modify the contents.
  Intermediate Node: A neighbor B that receives the message rebroadcasts the message after including its own cryptographic credentials. B signs the encrypted portion of the received SPC, includes its own certificate, and re-encrypts with the public key of X. This public key can be obtained in the certificate forwarded by A. $B \rightarrow broadcast: SPC, IP_X, Cert_X, [[[IP_X, Cert_A, N_A, t]K_A -]K_X +]K_B -, Cert_B]K_X +$ Nodes that receive the SPC packet create entries in their routing table so as not to forward duplicate packets. The entry also serves to route the reply packet from the destination along the reverse path.
  Destination Node: Once the destination X receives the SPC, it checks that all the signatures are valid. X replies to the first SPC it receives and also any SPC with a shorter recorded path. X sends a Recorded Shortest Path (RSP) message to the source through its predecessor D $X \rightarrow D: [RSP, IP_A, Cert_X, N_A, route]K_X -$.The source eventually receives the packet and verifies that the nonce corresponds to the SPC is originally generated.

Advantages: The onion-like signing of messages prevents nodes in the middle from changing the path in several ways. First, to increase the path length of the SPC, malicious nodes require an additional valid certificate. Second, malicious nodes cannot decrease the recorded path length or alter it because doing so would break the integrity of the encrypted data.

- **Route Maintenance:** ARAN is an on-demand protocol. Nodes keep track of whether routes are active [58]. When no traffic has occurred on an existing route for that route's lifetime, the route is simply de-activated in the route table. Data received on an inactive route causes nodes to generate an Error (ERR) message that travels the reverse path towards the source. Nodes also use ERR messages to report links in active routes that are broken due to node movement. All ERR message must be signed. For a route between source A and destination X, a node B generates the ERR message for its neighbor C as follows: $B \rightarrow C: [ERR, IP_A, IP_X, Cert_C, N_B, t]K_B$ This message is forwarded along the path towards the source without modification. A nonce and timestamp ensures the ERR message is fresh. Because messages are signed, malicious nodes cannot generate ERR messages for other nodes. The non-repudiation provided by the signed ERR message allows a node to be verified as the source of each ERR message that it sends. A node which transmits a large number of ERR messages, whether the ERR messages are valid or fabricated, should be avoided.

B. SEAD

Our Secure Efficient Ad hoc Distance vector routing protocol (SEAD) is robust against multiple uncoordinated attackers creating incorrect routing state in any other node, in spite of active attackers or compromised nodes in the network[50]. To support use of SEAD with nodes of limited CPU processing capability and to guard against DoS attacks in which an attacker attempts to cause other nodes to consume excess network bandwidth or processing time, we use efficient one-way hash functions

- **Hash chains:** A one-way hash chain is built on a one-way hash function [52][58]. Like a normal hash function, a one-way hash function *H* maps an input of any length to a fixed-length bit string. Thus, H: $\{0,1\}^* \rightarrow \{0,1\}^p$, where p. is the length in bits of the hash function's output. The function *H* should be simple to compute yet must be computationally infeasible in general to invert. To create a one-way hash chain, a node chooses a random $x \in \{0,1\}^p$ and computes the list of values h_0, h_1, h_2, h_3……h_n, where $h_0 = x$, and $h_i = H(h_i - 1)$ for $0 < i \leq n$, for some *n*. The node at initialization generates the elements of its hash chain using this recurrence, in order of increasing subscript *i*; over time, it uses certain elements of the chain to secure its routing updates. In using these values, the node progresses in order of decreasing subscript i within the generated chain. Given an existing authenticated element of a one-way hash chain, we can verify elements later in the sequence of use within the chain (further on, in order of decreasing subscript). For example, given an authenticated h_i value, a node can authenticate $h_i - 3$ by computing $H(H(H(h_i - 3)))$ and verifying that the resulting value equals h_i. To use one-way hash chains for authentication, we assume some mechanism for a node to distribute an authentic element such as h_n from its generated hash chain.

SEAD for authenticating an entry in a routing update uses the *sequence number* in that entry to determine a contiguous group of *m* elements from that destination node's hash chain, one element of which must be used to authenticate that routing update. The particular element from this group of elements that must be used to authenticate the entry is determined by the *metric* value being sent in that entry. Specifically, if a node's hash chain is the sequence of values h_0, h_1, h_2, h_3……h_n and *n* is divisible by *m*, then for a sequence number *i* in some routing update entry, let $k=n/m - i$. An element from the group of elements h_{km}, $h_{km}+1$,……h_{km+m-1} from this hash chain is used to authenticate the entry; if the metric value for this entry is *j*, $0 \leq j \leq m$, then the value h_{km+j} here is used to authenticate the routing update entry for that sequence number.

C. SORP

OSPF is a link state routing protocol used within one autonomous system (AS) or routing domain. It creates a global network topology in three which are following

- **Phase I**: Neighbor and Adjacency Establishment A router broadcasts periodically a Hello packet to discover its neighboring routers. After the neighboring routers establish connections, they synchronize their databases with each other through a Database Exchange Process.
- **Phase II**: Information Exchange by LSA Flooding A router assembles the link state information about its local neighborhood into a Link State Advertisement (LSA) and floods it to the whole network.
- **Phase III**: Calculate Shortest Route using Link State Database After a router collects all the link state information, it calculates a shortest path tree with itself as the root by using Dijkstra algorithm and forms a complete structure of routing in the network. OSPF divides an AS into groups of routers called *areas*.

A two level hierarchy among these areas is established, with the top level defined as the backbone area and the second level consisting of many areas attached to the backbone. Routers belonging to a single area are called *internal routers*. Routers that belong to more than one area are called Area Border Routers (ABR). All ABRs belong to the backbone and several of the routers, within an area or within the backbone, which exchange information with an external autonomous system, are known as Autonomous System Boundary Routers (ASBR). Security Strong Points of OSPF routing protocol, some inherent properties of OSPF make it very robust to failures and some attacks.

- **Flooding And Information Least Dependency:** As we mentioned above, OSPF uses flooding for the dissemination of LSAs. This makes sure that within the same *area* all the routers have the identical topological database. Even if a router goes down, other routers can still exchange their link state information provided that an alternate path exists. Furthermore the link state information propagated in the network is the raw message generated by the original router instead of the summarized information from neighbors,

which is the situation for distance vector routing. This makes it easy to protect the authenticity of the information.

- **Hierarchy Routing and Information Hiding:** OSPF is a two level routing protocol which are intra-area routing and inter-area routing. ABRs connect to backbone and exchange summarized area information. Since intra-area routing depends only on information from within that area, it is not vulnerable to problems out of the area. And problems in one area will not influence the intra-area routing of other areas and inter-area routing among other areas. So hierarchy routing has security advantage.

D. SRP

Secure Routing Protocol [4][13] (Lightweight Security for DSR), which we can use with DSR to design SRP as an extension header that is attached to ROUTE REQUEST and ROUTE REPLY packets. SRP doesn't attempt to secure ROUTE ERROR packets but instead delegates the route-maintenance function to the Secure Route Maintenance portion of the Secure Message Transmission protocol. SRP uses a sequence number in the REQUEST to ensure freshness, but this sequence number can only be checked at the target. SRP requires a security association only between communicating nodes and uses this security association just to authenticate ROUTE REQUESTS and ROUTE REPLYS through the use of message authentication codes. At the target, SRP can detect modification of the ROUTE REQUEST, and at the source, SRP can detect modification of the ROUTE REPLY.

Because SRP requires a security association only between communicating nodes, it uses extremely lightweight mechanisms to prevent other attacks. For example, to limit flooding, nodes record the rate at which each neighbor forwards ROUTE REQUEST packets and gives priority to REQUEST packets sent through neighbors that less frequently forward REQUEST packets.
SRP authenticates ROUTE REPLYS from intermediate nodes using shared group keys or digital signatures. When a node with a cached route shares a group key with (or can generate a digital signature verifiable by) the initiator of the REQUEST, it can use that group key to authenticate the REPLYS. The authenticator, which is either a message authentication code, computed using the group key or a signature is called the intermediate node reply token. The signature or MAC is computed over the cache REPLY.

E. SECURE AODV

The SecAODV [54] implements two concepts secure binding between IPv6 addresses and the independent of any trusted security service, Signed evidence produced by the originator of the message and signature verification by the destination, without any form of delegation of trust. The SecAODV implementation follows Tuominen's design which uses two kernel modules ip6_queue, ip6_nf_aodv, and a user space daemon AODV. The AODV daemon then generates a 1024-bit RSA key pair. Using the public key of this pair, the securely bound global and site-local IPv6 addresses are generated.

The AODV protocol is comprised of two basic mechanisms, route discovery and maintenance of local connectivity. The SecAODV protocol adds security features to the basic AODV mechanisms, but is otherwise identical. A source node that requests communication with another member of the MANET referred to as a destination D initiates the process by constructing and broadcasting a signed route request message RREQ. The format of the SecAODV RREQ message differs from the one proposed in [18], it additionally contains the RSA public key of the source node S and is digitally signed to ensure authenticity and integrity of the message. Upon receiving a RREQ message, each node authenticates the source S, by verifying the message integrity and by verifying the signature against the provided public key. Upon successful verification, the node updates its routing table with S's address and the forwarding node's address. If the message is not addressed to it, it rebroadcasts the RREQ.

F. BISS

Building Secure Routing out of an Incomplete Set of Security Associations (BISS) [38], the sender and the receiver can establish a secure route, even if, prior to the route discovery, only the

receiver has security associations established with all the nodes on the chosen route. Thus, the receiver will authenticate route nodes directly through security associations. The sender, however, will authenticate directly the nodes on the route with which it has security associations, and indirectly (by exchange of certificates) the node with which it does not have security associations. The operation of BISS ROUTE REQUEST relies on mechanisms similar to direct route authentication protocols. When an initiator sends a ROUTE REQUEST, it signs the request with its private key and includes its public key *PKI* in the request along with a certificate *cI* signed by the central authority binding its id with *PKI* .

This enables each node on the path to authenticate the initiator of the ROUTE REQUEST. The ROUTE REQUEST message contains the id of the target node. The node that receives this ROUTE REQUEST authenticates the initiator (by verifying the signature on the message), and tries to authenticate the target directly through security associations that it has. Only if a node can successfully authenticate both the initiator and the target will the node broadcast the message further. In BISS, we use similar route request data authentication mechanisms as in Ariadne.

G. SLSP

The Secure Link State Protocol (SLSP) [30] for mobile ad hoc networks is responsible for securing the discovery and distribution of link state information. The scope of SLSP may range from a secure neighborhood discovery to a network-wide secure link state protocol. SLSP nodes disseminate their link state updates and maintain topological information for the subset of network nodes within *R* hops, which is termed as their *zone* . Nevertheless, SLSP is a self-contained link state discovery protocol, even though it draws from, and naturally fits within, the concept of hybrid routing. To counter adversaries, SLSP protects link state update (*LSU*) packets from malicious alteration, as they propagate across the network.

It disallows advertisements of non-existent, fabricated links, stops nodes from masquerading their peers, strengthens the robustness of neighbor discovery, and thwarts deliberate floods of control traffic that exhausts network and node resources. To operate efficiently in the absence of a central key management, SLSP provides for each node to distribute its public key to nodes within its zone. Nodes periodically broadcast their certified key, so that the receiving nodes validate their subsequent link state updates. As the network topology changes, nodes learn the keys of nodes that move into their zone, thus keeping track of a relatively limited number of keys at every instance. SLSP defines a secure neighbor discovery that binds each node *V* to its Medium Access Control (*MAC*) address and its *IP* address, and allows all other nodes within transmission range to identify *V* unambiguously, given that they already have *EV*. Nodes advertise the state of their incident links by broadcasting periodically signed link state updates (*LSU*). SLSP restricts the propagation of the *LSU* packets within the zone of their origin node. Receiving nodes validate the updates, suppress duplicates, and relay previously unseen updates that have not already propagated *R* hops. Link state information acquired from validated *LSU* packets is accepted only if both nodes incident on each link advertise the same state of the link.

H. TIARA

Techniques for Intrusion-Resistant Ad Hoc Routing Algorithms (TIARA) mechanisms protect ad hoc networks against denial-of-service (DoS) attacks launched by malicious intruders. TIARA addresses two types of attacks on data traffic which are flow disruption and resource depletion. The innovation is following

- Routing algorithm independent approach for dealing with flow disruption and resource depletion attacks
- Fully distributed, self configuring firewall confines impact of DoS attack to immediate neighborhood of offending node
- Intrusion-resistant overlay routing reconfigures routes to circumvent malicious nodes

Wireless Router Extension implementation architecture enables TIARA survivability mechanisms to be easily incorporated within existing wireless IP routers.

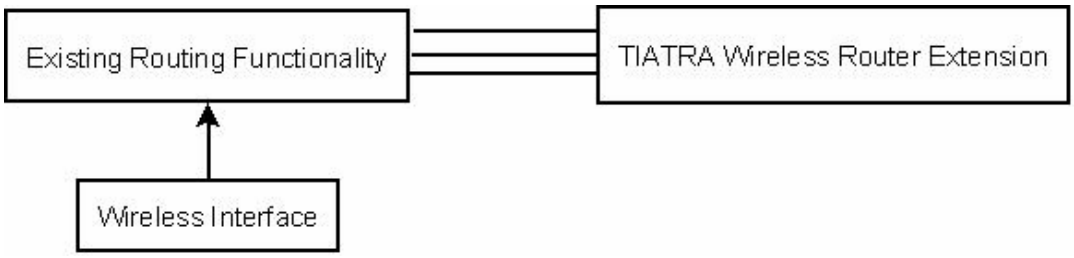

FIGURE 2: WIRELESS ROUTER EXTENSION IN TIARA

I. ARIADNE

A Secure On Demand Routing Protocol for Ad Hoc Networks (ARIADNE) using the TESLA [43][44] broadcast authentication protocol for authenticating routing messages, since TESLA is efficient and adds only a single message authentication code (MAC) to a message for broadcast authentication. Adding a MAC (computed with a shared key) to a message can provide secure authentication in point-to-point communication; for broadcast communication, however, multiple receivers need to know the MAC key for verification, which would also allow any receiver to forge packets and impersonate the sender. Secure broadcast authentication thus requires an asymmetric primitive, such that the sender can generate valid authentication information, but the receivers can only verify the authentication information. TESLA differs from traditional asymmetric protocols such as RSA in that TESLA achieves this asymmetry from clock synchronization and delayed key disclosure, rather than from computationally expensive one-way trapdoor functions.

Design and evaluation of Ariadne, a new ad hoc network routing protocol that provides security against one compromised node and arbitrary active attackers, and relies only on efficient *symmetric* cryptography [49]. Ariadne operates on-demand, dynamically discovering routes between nodes only as needed; the design is based on the basic operation of the DSR protocol. Rather than generously applying cryptography to an existing protocol to achieve security, however, we carefully re-designed each protocol message and its processing. The security mechanisms we designed are highly efficient and general, so that they should be applicable to securing a wide variety of routing protocols.

This article presents the TESLA (Timed Efficient Stream Loss-tolerant Authentication) broadcast authentication protocol, an efficient protocol with low communication and computation overhead, which scales to large numbers of receivers, and tolerates packet loss. TESLA is based on loose time synchronization between the sender and the receivers. TESLA broadcast authentication protocol have the following requirements: Low computation overhead for generation and Verification of authentication information. Low communication overhead is limited buffering required for the sender and the receiver, hence timely authentication for each individual packet which are Robustness to packet loss, Scales to a large number of receivers.

J. SAR

Security-Aware ad hoc Routing (SAR) that incorporates security attributes as parameters into ad hoc route discovery. SAR enables the use of security as a negotiable metric to improve the relevance of the routes discovered by ad hoc routing protocols. 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 [58]. 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.

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, use AODV[40] as our platform to implement SAR.

V. COMPARISONS OF SECURE PROTOCOLS

At the last we provide the comparison of different secure routing protocols of ad hoc network using table 1 and table 2. In table 1 shows defense against different type of attack. Comparison shows which protocol is better in different type of attacks. For example replay attack cover by ARAN but it is not coverable by RAP [58].

Attack	**Protocol**							
	ARAN	SRP	SEAD	ARIADEAN	SAODV	SLSP	OSRP	RAP
Location Disclosure	No	No	No	No	No	No	No	No
Black- Hole	No	No	No	No	No	No	Yes	No
Replay	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Worm hole	No	No	No	No	No	No	No	No
Black mail	NA	NA	NA	NA	NA	NA	NA	NA
Denial of services	No	Yes	Yes	Yes	No	Yes	No	No
Routing table poisoning	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Rushing attacks	Yes	No	Yes	Yes	No	No	No	Yes

Table 2: DEFENSE AGAINST ATTACK

Table 3 shows the proposed solution according to the requirement as well as shows the characteristics of different routing protocols for different operation parameter. Proposed solution describe for protocols used to provide security in adhoc and routing approach used here for adhoc routing protocols used in secure routing protocol. For loop freedom use protocol a sequence no which avoid the count infinite problem. Routing algorithms have used many different metrics to determine the best route. Sophisticated routing algorithms can base route selection on multiple metrics, combining them in a single metric. All the following metrics have been used: Path length, Reliability, Delay, Bandwidth, Load and Communication cost. The shortest path problem is the problem of finding a path between two nodes such that the sum of the cost of its constituent channel is minimized.

PROPOSED SOLUTON	ROUT-ING APPROACH	LOOP FREEDOM	ROUT-ING METRIC	SHORT-EST PATH	REPLY TO ROUTE REQUESTS	REQUIREMENTS
ARAN	On-demand	Yes	None	Optional	No	Online trusted certification authority.
SAR	On-demand	Depends on the selected Security requirement.	A security requirement	No	No	Key distribution or secret sharing mechanism.
SRP	On-demand	Yes	Distance	No	Optional	Existence of a security association between each source and destination node.
SEAD	Table-driven	Yes	Distance	No	No	Clock synchronization,
ARIADNE	On-demand	Yes	Distance	No	No	TESLA keys are distributed to the participating nodes via an online key distribution center.
SAODV	On-demand	Yes	Distance	No	Optional	Online key management scheme for the acquisition and verification of public keys.
TIARA	On-demand	Depends on the basis protocol	Distance	Depends on the basis protocol	Depends on the basis protocol	Online public key infrastructure.
SLSP	Table-driven	Yes	Distance	No	No	Nodes must have their public keys certified by a TTP
BISS	On-demand	Yes	Distance	No	No	The target node of a route discovery must share secret keys with all the intermediate nodes
IPsec	NA	NA	NA	NA	NA	Prearranged common secrets between each pair of nodes, or an online trusted third party

Table 3: OPERATIONAL REQUIREMENT AND PARAMETER FOR THE PROPOSED SOLUTION

VI. CONCLUSION

We have presented an overview of the existing security scenario in the Ad-Hoc network environment. Key management, Ad-hoc routing of wireless Ad-hoc networks were discussed. Ad-hoc networking is still a raw area of research as can be seen with the problems that exist in these networks and the emerging solutions. The key management protocols are still very expensive and not fail safe. Several protocols for routing in Ad-hoc networks have been proposed. There is a need to make them more secure and robust to adapt to the demanding requirements of these

networks. The flexibility, ease and speed with which these networks can be set up imply they will gain wider application. This leaves Ad-hoc networks wide open for research to meet these demanding application.

VII. REFERENCES

[1] Adrian Perrig Ran Canetti J. D. Tygar Dawn Song "*The TESLA Broadcast Authentication Protocol*", UC Berkeley and IBM Research.

[2] Ajay Mahimkar, R. K. Shyamasundar "*S-MECRA A Secure Energy-Efficient Routing Protocol for Wireless Ad Hoc Networks*" IEEE 2004

[3] Alia Fourati, Khaldoun Al Agha, Hella Kaffel Ben Ayed "Secure and Fair Auctions over Ad Hoc Networks" *Int. J. Electronic Business, 2007*

[4] Anand Patwardhan, Jim Parker, Michaela Iorga. Anupam Joshi, "*Tom Karygiannis, Secure Routing and Intrusion Detection in Ad Hoc Networks*" 3rd International Conference on Pervasive Computing and Communications (PerCom 2005), Kauai Island, Hawaii**.**

[5] Bing Wua, Jie Wua, Eduardo B. Fernandeza, Mohammad Ilyasa, Spyros Magliveras, "*Secure and efficient key management in mobile ad hoc networks*" Journal of Network and Computer Applications 30 (2007) 937–954

[6] Bissias, G.D., Liberatore, M., Jensen, D., Levine, B.N., "Privacy vulnerabilities in encrypted HTTP streams" In *Proc. Privacy Enhancing Technologies Workshop* (PET 2005).

[7] C. E. Perkins, E. M. Royer, and S. R. Das, "*Ad Hoc On-Demand Distance Vector (AODV) Routing*," IETF Mobile Ad Hoc Networks Working Group, Internet Draft, work in progress, 17 February 2003.

[8] F. Hu and N. K. Sharma, "*Security Considerations in Ad Hoc Networks*," to be appeared in Ad Hon Network, 2004.

[9] F. Anjum*,* Anup K. Ghosh*,* nada golmie*,* paul kolodzy*,* radha poovendran*,* rajeev shorey*,* d. Lee*, j-sac, "Security in Wireless Ad hoc Networks*", ieee journal on selected areas in communications, vol. 24, no. 2, February 2006.

[10] H.-A. Wen, C.-L. Lin, and T. Hwang, "*Provably Secure Authenticated Key Exchange Protocols for Low Power Computing Clients*," Computers and Security, vol. 25, pp. 106-113, 2006.

[11] Haiyun Luo, Petros Zerfos, Jiejun Kong, Songwu Lu, Lixia Zhang, "*Self-securing Ad Hoc Wireless Networks*", 7th IEEE Symp. on Comp. and Communications (ISCC), Taormina, 2002.

[12] Hongmei Deng, Wei Li, and Dharma P. Agrawal, "Routing Security in Wireless Ad Hoc Networks", IEEE Communications Magazine October 2002.

[13] Huaizhi Li Zhenliu Chen Xiangyang Qin, "*Secure Routing in Wired Networks and Wireless Ad Hoc Networks" IEEE, 2004.*

[14] Huaizhi Li, Mukesh Singha, *"Trust Management in Distributed Systems"* IEEE Computer Society February 2007.

[15] I. Aad, J.-P. Hubaux, and E-W. Knightly, *"Denial of Service Resilience in Ad Hoc Networks,"* Proc. MobiCom, 2004.

[16] J. Nam, S. Cho, S. Kim, and D. Won, *"Simple and Efficient Group Key Agreement Based on Factoring"* Proc. Int'l Conf. Computational Science and Its Applications (ICCSA '04), pp. 645-654, 2004.

[17] J. Parker, J. L. Undercoffer, J. Pinkston, and A. Joshi., "*On Intrusion Detection in Mobile Ad Hoc Networks"*. In 23rd IEEE International Performance Computing and Communications Conference Workshop on Information Assurance. IEEE, April 2004.

[18] Jeremy J. Blum, Member, IEEE, and Azim Eskandarian, Member, IEEE, "*A Reliable Link-Layer Protocol for Robust and Scalable Intervehicle Communications"* IEEE Transactions On Intelligent Transportation Systems, vol. 8, no. 1, March 2007.

[19] Jung-San Lee, Chin-Chen Chang, "*Secure communications for cluster-based ad hoc networks using node identities"* Journal of Network and Computer Applications 22 October 2006

[20] K. Balakrishnan, J. Deng, and P.K. Varshney, "*TWOACK: Preventing Selfishness in Mobile Ad Hoc Networks"* Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '05), Mar. 2005.
[21] Karan Singh, Rama Shankar Yadav, Raghav Yadav, R. Shiva Kumaran, *"Adaptive Multicast Congestion Control "* HIT haldia March 2007.
[22] Kejun Liu, Jing Deng, Member, IEEE, Pramod K. Varshney, Fellow, IEEE, and Kashyap Balakrishnan, Member, IEEE, "*An Acknowledgment-Based Approach for the Detection of Routing Misbehavior in MANETs"* IEEE Transaction on Mobile Computing, VOL. 6, NO. 5, May 2007
[23] L. Buttyan and J.-P. Hubaux, "*Security and Cooperation in Wireless Networks*," http://secowinet.epfl.ch/, 2006.
[24] M. Bechler, H.-J. Hof, D. Kraft, F. Pählke, L. Wolf, "A Cluster-Based Security Architecture for Ad Hoc Networks" IEEE INFOCOM 2004.
[25] Mike Just_ Evangelos Kranakis Tao Wan, *"Resisting Malicious Packet Dropping in Wireless Ad Hoc Networks"* Internet draft: draft-ietfitrace-03.txt, January 2003.
[26] Mohammad Al-Shurman and Seong-Moo Yoo, Seungjin Park, *"Black Hole Attack in Mobile Ad Hoc Networks" ACMSE'04*, April 2-3, 2004, Huntsville, AL, USA.
[27] Muhammad Bohio, Ali Miri, E.cient, "*Identity-based security schemes for ad hoc network routing protocols*" Ad Hoc Networks 2 (2004) 309–317
[28] Nikos Komninos, Dimitris Vergados, Christos Douligeris, "*Layered security design for mobile ad hoc networks"* journal computers & security 25, 2006 , pp. 121 – 130.
[29] Nobuo Okabe, Shoichi Sakane, Kazunori Miyazawa, Ken'ichi Kamada, "*Extending a Secure Autonomous Bootstrap Mechanism to Multicast Security*" 2007 International Symposium on Applications and the Internet Workshops (SAINTW'07).
[30] P. Papadimitratos and Z.J. Haas, "*Secure Link State Routing for Mobile Ad Hoc Networks*" Proc. IEEE Workshop on Security and Assurance in Ad Hoc Networks, IEEE Press, 2003, pp. 27–31.
[31] Panagiotis Papadimitratos , Zygmunt J. Haas, "*Secure message transmission in mobile ad hoc networks, Ad Hoc Networks*" IEEE 2003, 193–209 .
[32] R. Hinden and S. Deering. RFC 3513, "*Internet Protocol Version 6 (IPv6) Addressing Architecture*" April 2003.
[33] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "*Sustaining Cooperation in Multi-Hop Wireless Networks*," Proc. Second Symp. Networked Systems Design and Implementation, Apr. 2005.
[34] R. Shiva Kumaran, Rama Shankar Yadav, Karan Singh "*Multihop wireless LAN* " HIT haldia March 2007.
[35] *S. Holeman, G. Manimaran, J. Davis, A. Chakrabarti, Differentially secure multicasting and its implementation methods,* Computers & Security Vol 21, No 8, pp736-749, 2002.
[36] S.M. Bellovin, M. Leech, and T. Taylor. ICMP Traceback Messages. Internet draft: draft-ietfitrace 03.txt, January 2003.
[37] Seung Yi, Prasad Naldurg, Robin Kravets, "*A Security-Aware Routing Protocol for Wireless Ad Hoc Networks*" IEEE 2003.
[38] Srdjan Capkun and Jean-Pierre Hubaux, "*Building Secure Routing out of an Incomplete Set of Security Associations" WiSE'03,* September 19, 2003, San Diego, California, USA.
[39] Stallings, W., *Wireless Communications and Networks*, 2nd Ed., Prentice Hall, 2005.
[40] T. Aura. Internet Draft: Cryptographically Generated Addresses (CGA). http://www.ietf.org/proceedings/ 04mar/I-D/draftietf- send-cga-05.txt, February 2004.
[41] Thomas S. Messerges, ohnas Cukier, Tom A.M. Kevenaar, Larry Puhl, Rene truik, Ed Callaway, *"A Security Design for a General Purpose, Self-Organizing, Multihop Ad Hoc Wireless Network*" 1st ACM Workshop Security of Ad Hoc and Sensor Networks Fairfax, Virginia 2003
[42] Tzonelih Hwang, Kuo-Chang Lee, Chuan-Ming Li, "*Provably Secure Three-Party Authenticated Quantum Key Distribution Protocols*" IEEE Transactions On Dependable And Secure Computing, vol. 4, no. 1, January-March 2007.
[43] Uppsala University, The Ad hoc Protocol Evaluation (APE) test bed, release 0.3, downloaded Nov. 2005.

[44] Uppsala University, The AODV-UU implementation version 0.8.1, downloaded Nov. 2005.
[45] W. Xu, T. Wu, "*TCP Issues in Mobile Ad Hoc Networks: Challenges and Solutions*", Journal of Computer Science and Technology, 2006, 21.
[46] Weiqiang Xu and Tiejun Wu, "A Congestion Control Algorithm for Ad Hoc Networks: A Dual Decomposition Approach" 6th World Congress on Intelligent Control and Automation, June 21 - 23, 2006, Dalian, China.
[47] Wright, C.V., Monrose, F., Masson, G.M., "*HMM profiles for network traffic classification*" in *Proc.* ACM Workshop on Visualization and Data Mining for Computer Security, pp. 9–15, Oct. 2004.
[48] Wright, C.V., Monrose, F., Masson, G.M., "*Towards better protocol identification using profile HMMs"* JHU Technical Report JHU-SPAR051201, 14p., June, 2005.
[49] Y. Xue and K. Nahrstedt, "*Providing Fault-Tolerant Ad-Hoc Routing Service in Adversarial Environments,"* Wireless Personal Comm., vol. 29, nos. 3-4, pp. 367-388, 2004.
[50] Y.-C. Hu, D. B. Johnson, and A. Perrig., *"SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks"* In Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications, page 3. IEEE Computer Society, 2002.
[51] Yang Mingxi, Li Layuan, Fang Yiwei, "Securing multicast route discovery for mobile ad hoc networks" SpringerLink, February 17, 2007.
[52] Yih-chun hu, adrian perrig, "*A Survey of Secure Wireless ad hoc routing*" IEEE security & privacy May-June 2004
[53] Yih-Chun Hu, Adrian Perrig, and David B. Johnson., "*Packet Leashes A Defense against Wormhole Attacks in Wireless Ad Hoc Networks*" In Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003), April 2003. To appear.
[54] Yih-Chun Hu, Adrian Perrig, David B. Johnson Ariadne: "*A Secure On-Demand Routing Protocol for Ad Hoc Networks*" *MobiCom'02,* September 23–26, 2002, Atlanta, Georgia, USA.
[55] Yih-Chun Hu, Adrian Perrig, David B. Johnson, "Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols" *WiSe 2003,* September 19, 2003, San Diego, California, USA.
[56] Yuh-Ren Tsai, Shiuh-Jeng Wang, *"Routing Security and Authentication Mechanism for Mobile Ad Hoc Networks"* Chung-Shan Institute of Science and Technology, Taiwan, R.O.C., under Grant BC-93-B14P and the National Science Council, Taiwan, R.O.C., IEEE 2004.
[57] S.W. Smith, , "*A Case (Study) For Usability in Secure Email Communication*" IEEE Computer Society 2007
[58] Patroklos g. Argyroudis and donal o'mahony, "*Secure Routing for Mobile Ad hoc Networks*", IEEE Communications Surveys & Tutorials Third Quarter 2005.