INTERNET-DRAFT Panagiotis Papadimitratos, Cornell University
Zygmunt J. Haas, Cornell University
Prince Samar, Cornell University

Expires in six months on March 2003 September 2002

The Secure Routing Protocol (SRP) for Ad Hoc Networks
<draft-secure-routing-protocol-srp-00.txt>

Status of this Memo

Abstract

This document describes the Secure Routing Protocol (SRP), a route discovery protocol for ad hoc networks that mitigates the detrimental effects of maliciously behaving nodes that disrupt the route discovery in order to obstruct or disable the network operation. Our protocol provides correct routing information; i.e., factual, up-to-date and authentic connectivity information regarding a pair of nodes that wish to communicate in a secure manner. The sole requirement is that any two such end nodes have a security association. Accordingly, SRP does not require that any of the intermediate nodes perform cryptographic operations or have a prior association with the end nodes. The end-to-end operation of SRP allows for efficient cryptographic mechanisms, such as message authentication codes. More importantly, SRP can be used in a wide range of MANET instances, without restrictive assumptions on the underlying trust, network size and membership.

Contents

Status of this Memo i
Abstract i

Applicability Statement iii
A. Networking Context iii
B. Protocol Characteristics and Mechanisms iii

1. Introduction 1

2. The Secure Routing Protocol (SRP) 2
2.1 Assumptions . . . . . .
2.2 Overview . . . . . .

3. Protocol Description . . . . . .
3.1 The Neighbor Lookup Protocol (NLP) . . . . . .
3.2 Route Request . . . . . .
3.3 Query Handling . . . . . .
3.4 Route Reply . . . . . .
3.5 SRP Extension . . . . . .
3.6 Route Maintenance . . . . . .

4. Implementation Details . . . . . .
4.1 Packet Formats . . . . . .
4.2 Data Structures . . . . . .

5. References . . . . . .

Authors' Information . . . . . .
MANET Contact Information . . . . . .

Applicability Statement

A. Networking Context

The Secure Routing Protocol is designed to operate over a wide variety of ad hoc networks, under the least restrictive assumptions. SRP does not rely on any assumption on the node mobility, node equipment (such as Global Positioning System (GPS)), and network size. Furthermore, SRP does not make any assumption on the network membership and trust, and does not require that each node be able to present and validate credentials, such as public keys, for all other network nodes. Finally, SRP does not rely on intrusion detection or monitoring techniques and does not assume any regularity or patterns of malicious behavior in order to identify and isolate adversarial nodes; instead, SRP is capable of operating in the presence of adversaries that actively disrupt the route discovery.

B. Protocol Characteristics and Mechanisms

* Does the protocol provide support for unidirectional links? (If so, how?)

  No, only bi-directional links are used (as in 802.11).

* Does the protocol require the use of tunneling? (If so, how?)

  No.

* Does the protocol require using some form of source routing? (If so, how?)

  Yes, currently, SRP utilizes source routing; in the future, we will present the protocol’s operation without the use of source routing.

* Does the protocol require the use of periodic messaging? (If so, how?)

  No.

* Does the protocol require the use of reliable or sequenced packet delivery? (If so, how?)

  No.

* Does the protocol provide support for routing through a multi-technology routing fabric? (If so, how?)

  Yes. It is assumed that each node's network interface is assigned a single IP address.

* Does the protocol provide support for multiple hosts per router? (If so, how?)

  No. SRP supports only a single host per router. However, SRP can support multiple hosts per router in the following case: Hosts operate in logically separated but physically co-located overlaid domains.

* Does the protocol support the IP addressing architecture? (If so, how?)

Yes. Each node is assumed to have a single IP address. The SRP references all nodes by their IP address.

* Does the protocol require link or neighbor status sensing (If so, how?)

Yes. Each node maintains a valid and consistent mapping of the Medium Access Control and the IP layer addresses of used by its neighbors, as extracted from received (or overheard) frames.

* Does the protocol have dependence on a central entity? (If so, how?)

No. The SRP is a fully distributed protocol.

* Does the protocol function reactively? (If so, how?)

Yes. Route queries are initiated by nodes intending to send packets to the destination nodes.

* Does the protocol function proactively? (If so, how?)

No.

* Does the protocol provide loop-free routing? (If so, how?)

Yes. Routes are inherently loop-free, as source routing is used.

* Does the protocol provide for sleep period operation? (If so, how?)

Yes. SRP operates correctly even if nodes go into and out of sleep mode at arbitrary times.

* Does the protocol provide some form of security? (If so, how?)

Yes. SRP guarantees the discovery of correct connectivity information in a timely manner, over an unknown network, in the presence of malicious nodes that disrupt the route discovery operation.

1. Introduction

The provision of security services in the Mobile Ad Hoc Networks (MANET) context faces a set of challenges specific to this new technology. The insecurity of the wireless links, energy constraints, relatively poor physical protection of nodes in a hostile environment, the vulnerability of statically configured security schemes, and the absence of a fixed infrastructure have been identified [4,5] in literature as such challenges. The secure operation of the routing protocol is especially important for self-organizing MANET infrastructures, envisioned to operate in an open, collaborative, and highly volatile environment. There is no guarantee that such an environment will be free of malicious nodes, which do not comply with the employed protocol and attempt to harm the network operation.

The mechanisms currently incorporated in MANET routing protocols cannot cope with disruptions due to malicious node behavior. For example, any node could claim that it is one hop away from the sought destination, causing all routes to the destination to pass through itself. Or, a malicious node could corrupt any in-transit route request or reply packet and cause data to be misrouted. Moreover, adversaries could deny communication by “flooding” forged routing traffic (e.g., route queries or link state updates), falsely appearing as the most up-to-date information originating from other network nodes and, thus, cause their subsequent legitimate updates to be disregarded. Such malicious behavior could disable the network operation altogether, or incur long delays due to repeated attempts of route discovery before any two nodes be able to communicate.

The Secure Routing Protocol (SRP) [1] safeguards the acquisition of topological information by countering attacks that disrupt or exploit the route discovery operation to deny communication. Our protocol departs from solutions presented in the context of wire-line Internet [2], which require the existence of a trust structure that encompasses all nodes participating in routing, and may rely on network management operations to detect routing instabilities.

The novelty of SRP lies in that the correctness of the discovered route(s) can be verified from the route "geometry" itself. At the same time, false or corrupted control traffic is discarded in parts by the end nodes, thanks to the end-to-end security association, and in parts by the intermediate benign nodes, without cryptographic processing in the latter case. Basically, route requests propagate verifiably to the sought trusted destination and route replies are returned strictly over the reversed route, as accumulated in the route request packet. Moreover, intermediate nodes do not relay route replies unless their downstream node had previously relayed the corresponding query. In order to guarantee this crucially important functionality, the interaction of the protocol with the IP-related functionality is explicitly defined. An intact reply implies that (i) the reported path is the one placed in the reply packet by the destination, and (ii) the corresponding connectivity information is correct, since the reply was relayed along the reverse of the discovered route and consists of all nodes that participated in both phases of the route discovery.

The securing of the route discovery deprives the adversarial nodes of an "effective" means to systematically disrupt the communications of their peers. Despite our minimal trust assumptions, attackers cannot impersonate the destination and redirect data traffic, cannot respond with stale or corrupted routing information, are prevented from broadcasting forged control packets to obstruct the later propagation of legitimate queries, and are unable to influence the topological knowledge of benign nodes. To that extent, SRP provides very strong assurances on the correctness of the link-level connectivity information as well. It precludes adversarial nodes from controlling multiple potential routes per source-destination pair, and from forming “dumb” relays, that is, from not placing themselves in a route whose discovery they assisted.

The security features of SRP do not undermine its efficiency, that is, the ability of nodes to quickly respond to topological changes and discover correct routes. On the other hand, the protocol retains its ability to operate when under attack, with adversaries actively disrupting the route discovery. Moreover, the low processing overhead, especially due to cryptographic operations, renders SRP applicable for nodes with limited computational resources. Finally, the reliance on the basic and widely accepted reactive route discovery mechanism (broadcasted route query packets traverse the network as the relaying intermediate nodes append their identifier (IP address)) allows SRP to naturally extend a number of existing protocols. In particular, the IERP [13] of the Zone Routing Protocol (ZRP) [14] framework, the Dynamic Source Routing (DSR) [8], and ABR [15] are protocols that can incorporate the features of SRP with minimal or limited modifications.

2. The Secure Routing Protocol (SRP)

2.1 Assumptions

SRP focuses on (bi-directional) communication between a pair of nodes. A Security Association (SA) MUST exist between the source node S and the destination node T. Such an association could be instantiated, for example, by the knowledge of the public key of the other communicating end. The two nodes can negotiate a shared secret key, e.g., via the Elliptic Curve Diffie-Hellman algorithm [7,12], and then, using the SA, verify that the principal that participated in the exchange was indeed the trusted node. The existence of the SA is justified, because the end hosts choose to employ a secure communication scheme and, consequently, should be able to authenticate each other. However, the existence of SA's with any of the intermediate nodes is unnecessary. For the rest of the discussion, we assume the existence of a shared secret key K(S,T). The SA is bi-directional in that the shared key can be used for control traffic flowing in both directions, with relevant state maintained for each direction and end nodes able to use static or non-volatile memory.

The adversarial nodes may attempt to compromise the route discovery operation by exhibiting arbitrary, Byzantine behavior [3]. They are able to corrupt, replay, and fabricate routing packets, and capable of misrouting any packet in any possible manner. However, adversaries are also subject to the limitations of the communication environment, i.e., packet loss, path breakages etc, and have finite processing power.

The underlying data link layer (e.g., IEEE 802.11 [6]) provides reliable link transmission, without any requirement of data link security services, such as the Wired Equivalent Protocol (WEP) function. Moreover, links are assumed to be bi-directional, a requirement fulfilled by most of the proposed medium access control protocols, especially the ones employing the RTS/CTS dialogue. Moreover, due to the broadcast nature of the radio channel, each transmission is received by all neighbors, which are assumed to operate in promiscuous mode. Finally, it is expected that a one-to-one mapping between Medium Access Control and IP addresses exists. Nodes MAY select an arbitrary random IP address [10], or IP addresses MAY be assigned dynamically as roaming nodes join MANET domains. Nevertheless, each node has a single network interface at the data link layer.

2.2 Overview

The Secure Routing Protocol (SRP) safeguards the route discovery, requiring that only the end communicating nodes are securely associated, with no need for cryptographic operations on control traffic at intermediate nodes, two factors that render the scheme efficient and scalable. SRP places the overhead on the end nodes, an appropriate choice for a highly decentralized environment, and contributes to the robustness and flexibility of the scheme.

The source node S initiates the route discovery, by constructing a route request packet identified by a pair of identifiers: a query sequence number and a random query identifier. The source and destination and the unique (with respect to the pair of end nodes) query identifiers are the input for the calculation of the Message Authentication Code (MAC) [9], along with K(S,T). Route requests are (re-) broadcasted, while the identities (IP addresses) of the traversed intermediate nodes are accumulated in the route request packet.

Nodes maintain a limited amount of information identifying relayed request packets, so that packets that correspond to recent previously seen requests can be discarded. In addition, nodes maintain information regarding the data link and network addresses of their immediate neighbors, and perform a number of simple non-cryptographic checks on the relayed control traffic, based solely on the packet content, and discard non-compliant packets. Intermediate nodes MAY also regulate the service rate they provide to control traffic originating or being forwarded by each neighbor. Finally, they MAY provide the source of a route with a notification in the event of a path breakage, and MAY provide route replies, as explained in the “SRP Extension” section.

The destination T validates incoming request packets, and constructs route replies to not previously received queries originating from S. T calculates a MAC covering the route reply contents and returns the packet to S over the reverse of the route accumulated in the corresponding request packet. The destination MAY respond to more than one request packets of the same query, so that it provides the source with an as diverse topology picture as possible.

3. Protocol Description

The Secure Routing Protocol (SRP) introduces a set of new features to counter a wide range of attacks against the route discovery and guarantee the acquisition of correct connectivity information. We present here the functionality of SRP independently of how SRP can extend existing routing protocols.

3.1 The Neighbor Lookup Protocol

The Neighbor Lookup Protocol (NLP) is an integral part of SRP responsible for the following tasks: (i) It maintains a mapping of Medium Access Control and IP layer addresses of the node's neighbors, (ii) it identifies potential discrepancies, such as the use of multiple IP addresses by a single data-link interface, and (iii) measures the rates at which control packets are received from each neighbor, by differentiating the traffic primarily based on Medium Access Control addresses. The measured rates of incoming control packets are provided to the routing protocol as well. This way control traffic originating from nodes that selfishly or maliciously attempt to overload the network can be discarded (Section 3.3).

Basically, NLP extracts and retains the 48-bit hardware source address for each received (overheard) frame along with the encapsulated IP address. This requires a simple modification of the device driver [18], so that the data link address is “passed up” to the routing protocol with each packet. With nodes operating in promiscuous mode, the extraction of such pairs of addresses from all overheard packets leads to a reduction in the use of the neighbor discovery and query/reply mechanisms for medium access control address resolution. Each node updates its NEIGHBOR TABLE by retaining both addresses.

The mappings between data-link and network interface addresses are retained in the table as long as transmissions from the corresponding neighboring nodes are overheard. Each entry is associated with a NEIGHHBOR_LOST_TIMEOUT period and is removed from the table upon expiration. NEIGHHBOR_LOST_TIMEOUT should be greater than the timeout periods associated with the route discovery, such as the maximum delay before a new query is broadcasted.

NLP issues a notification to SRP in the event that according to the content of a received packet: (i) a neighbor used an IP address different from the address currently recorded in the neighbor table, (ii) two neighbors used the same IP address (that is, a packet appears to originate from a node that may have "spoofed" an IP address), (iii) a node uses the same medium access control address as the detecting node (in that case, the data link address may be “spoofed”). Upon reception of the notification, the routing protocol discards the packet bearing the address that violated the aforementioned policies.

Even though NLP does not rely on cryptographic validation, it thwarts adversaries from presenting themselves at the routing layer as more than one node. This would have been possible if different IP addresses were inserted in or used as the source address of the control traffic the adversary relays or originates. However, the effectiveness of NLP relies on the fact that medium access control addresses may be changed with substantial latency. NLP can be a significant line of defense, deterring, for example, a malicious node from flooding the network with spurious traffic. In any case, we should note that it is not of interest for SRP whether a relay node indeed presented itself with its 'actual' IP address, but whether the node participated in the discovery of the route.

3.2 Route request generation

A source node S maintains a query sequence number Q_SEQ for each destination it securely communicates with. The 32-bit Q_SEQ increases monotonically, for each request generated by S, and allows T to detect outdated route requests. The sequence number is initialized at the establishment of the SA and although it is not allowed to wrap around, it provides approximately a space of four billion query requests per destination. If the entire space is used, a new security association has to be established.

For each outgoing ROUTE_REQUEST, S generates a 32-bit random Query Identifier Q_ID, which is used by intermediate nodes as a means to identify the request. Q_ID is the output of a secure pseudorandom number generator [11]; its output is statistically indistinguishable from a truly random one and is unpredictable by an adversary with limited computational power. Since intermediate nodes have limited memory of past queries, uniqueness and randomness can be efficiently achieved, by using a one-way function (e.g., SHA-1 [16]) and a small random seed as input. This renders the prediction of the query identifiers practically impossible, and combats the following attack: malicious nodes simply broadcast fabricated requests only to cause subsequent legitimate queries to be dropped.

Along with Q_ID and Q_SEQ, the ROUTE_REQUEST header MUST include a Message Authentication Code (MAC). The MAC is a 96-bit long field, generated by a keyed hash algorithm [9], which calculates the truncated output of a one-way or hash function (e.g., SHA-1 or MD5 [17]). The one-way function input is the entire IP header, the basis protocol route request packet and most importantly, the shared key K(S,T). The Route Request fields that are updated as the packet propagates towards the destination, i.e., the accumulated addresses of the intermediate nodes, and the IP-header mutable fields are excluded.

The querying node MAY set the N_RREP field of the ROUTE_REQUEST header to indicate the number of route replies per query the destination SHOULD return. The default value for N_RREP is one (1). The source MAY increase N_RREP in case of a failed route discovery or in order to enrich its view of the network topology. Finally, in the case of a failed route discovery, the querying node SHOULD NOT generate a new ROUTE_REQUEST for the same destination before a period of WAIT_ROUTE_REPLY seconds. In the case of communication with K destinations, the overall rate of generating ROUTE_REQUEST packets SHOULD NOT exceed K/WAIT_ROUTE_REPLY queries per second.

3.3 Route Request Processing

Nodes receiving a ROUTE REQUEST parse the packet in order to determine whether an SRP header is present. If the SRP header is not present the packet MUST be dropped. Intermediate nodes extract the Q_ID value to determine if they have already relayed a packet corresponding to the same request. If not, they compare the last entry in the accumulated route to the IP datagram source address, which belongs to the neighboring node that relayed the request. The ROUTE REQUEST packet is dropped in the case of a mismatch or an NLP notification that the relaying neighbor violated one of the enforced policies. Otherwise, the packet is relayed (re-broadcasted), with the intermediate node inserting its IP address. The Q_ID, source and destination address field values are placed in the query table. Finally, intermediate nodes retain the IP addresses of their neighbors overheard forwarding (re-broadcasting) the query, in a FORWARD_LIST associated with the query table.

If the node is the sought destination T, the route request MUST be validated if T has a security binding with the querying node; otherwise, the packet is discarded. First, Q_SEQ is compared to S_MAX(S), the latest (highest) query sequence number received from S, within the lifetime of the S-T SA. If Q_SEQ < S_MAX(S), the request is discarded as outdated or replayed. If Q_SEQ = S_MAX(S) and T has already responded to a valid request, i.e., generated a route reply (in general, N_RREP replies), the request is disregarded.

Otherwise, T calculates the keyed hash of the request header and verifies its integrity and the authenticity of origin of the request packet. If validated, S_MAX(S) is set equal to max{Q_SEQ,S_MAX(S)} and a route reply is generated, as described in section 3.4.

In order to guarantee the responsiveness of the routing protocol, nodes maintain a priority ranking of their neighbors according to the rate of queries observed by NLP. The highest priority is assigned to the nodes generating (or relaying) requests with the lowest rate and vice versa. Quanta are allocated proportionally to the priorities and not serviced low-priority queries are eventually discarded. Within each class, queries are serviced in a round-robin manner.

Selfish or malicious nodes that broadcast requests at a very high rate are throttled back, first by their immediate neighbors and then by nodes farther from the source of potential misbehavior. On the other hand, non-malicious queries, that is, queries originating from benign nodes that regulate in a non-selfish manner the rate of their query generation, will be affected only for a period equal to the time it takes to update the priority (weight) assigned to a misbehaving neighbor. In the mean time, the round robin servicing of requests provides the assurance that benign requests will be relayed even amidst a "storm" of malicious or extraneous requests.

3.4 Route reply generation and forwarding

The destination generates one or more replies to each query. The number of replies does not exceed the min{N_RREP, NUMBER_OF_NEIGHBORS}. This restriction deters a malicious neighbor from relaying and having more than one ROUTE_REQUEST packets replied, and thus possibly control more than one route.

The ROUTE_REPLY is identified by the values of Q_SEQ and Q_ID of the corresponding ROUTE_REQUEST. The reverse of the route accumulated in the request packet is used as the source route of the ROUTE_REPLY packet. The destination MUST calculate, using K(S,T), and append a MAC covering the entire SRP header and the source route of the reply packet. The ROUTE_REPLY MUST be routed strictly along the reverse of the discovered route. This way, the source is provided with evidence that the request had reached the destination and that the reply was indeed returned along the reverse of the discovered route.

As the reply propagates along the reverse route, each intermediate relaying node MUST check whether the source address of the ROUTE_REPLY datagram is the same as the address of its downstream node, as reported in the ROUTE_REPLY. If not, or if and NLP notification has been received, the reply packet is discarded. The intermediate node MUST discard the reply if the corresponding request is not previously received and relayed.

Also, the reply packet MUST be discarded if it originates from a node that is not listed in FORWARD_LIST. This last control practically eliminates the possibility that a malicious node forms a “dumb” or “Byzantine” relay, even if it could change its medium access control address with insignificant delay and by-pass the defense provided by NLP. A “dumb” relay could have been formed if a node did not place its IP address in the ROUTE_REQUEST and relayed the ROUTE_REPLY without being listed in the discovered route [19], that is, it changed its data link and IP addresses as it relayed the request/reply packets to impersonate the previous relay without appearing in the route discovery.

Ultimately, the source validates the reply: it first checks whether it corresponds to a pending query. Then, it suffices to validate the MAC, and extract the route from the IP source route of the ROUTE_REPLY, which already provides the (reversed) discovered route.

3.5 The SRP Extension

The basic operation of SRP can be extended in order to allow for nodes, other than the destination, to provide route replies or feedback on the status of utilized routes. This may be possible if a subset of nodes share a common objective, belong to the same group G and mutually trust all the group members. In that case, the mutual trust could be instantiated by all group members sharing a secret key K(G).

Under this assumption, a querying node SHOULD append to each query an additional MAC calculated with the group key K(G), which we call Intermediate Node Reply Token (INRT). The functionality of SRP remains as described above, with the following addition: each group member maintains the latest query identifier seen from each of its peers, and can thus validate both the freshness and origin authenticity of queries generated from other group nodes.

Nodes other than the sought destination SHOULD respond to a validated request, if they have knowledge of a route to the destination in question. The ROUTE_REPLY is generated as above, except for the MAC calculation that uses K(G). The correctness of such a route is conditional upon the correctness of the information provided by the intermediate node, regarding the second portion of the route. When the ROUTE_REPLY is generated by the destination, an additional MAC(K(G),ROUTE_REPLY) SHOULD be appended apart from the MAC(K(S,T),ROUTE_REPLY). This would allow an intermediate node V that is part of the route and a member of G to utilize the discovered route suffix (i.e., the V to T part).

The INRT functionality can be provided independently from and in parallel with the one relying solely on the end-to-end security associations. For example, it could be useful for frequent intra-group communication; any two members can benefit from the assistance of their trusted peers, which may already have useful routes. Finally, the shared K(G) can be utilized for purposes that are beyond the discovery of routes. One example is the authentication of ROUTE_ERROR messages, as explained in section 3.6.

3.6 Route Maintenance

A ROUTE_ERROR packet SHOULD be generated by an intermediate node that fails to deliver a data packet to the next hop. ROUTE_ERROR packets MUST be source-routed to the source node S along the prefix of the route being reported as broken. The intermediate upstream nodes, with respect to the point of breakage, MUST check if the source address of the ROUTE_ERROR datagram is the same as the one of their downstream node, as reported in the broken route.

If there is no NLP notification that the relaying neighbor violated one of the enforced policies, the packet is relayed towards the source. In this case, NLP prevents an adversary that does not belong to but lies at a one-hop distance from the route from generating an error message. In such case, an inconsistency with the addresses already used (during the route discovery) by the actual downstream neighbor will be detected. The end node MUST compare the source-route of the error message to the prefix of the corresponding active route. This way, it verifies that the provided route error message refers to the actual route, and that it is not generated by a node that is not part of the route.

The correctness of the feedback (i.e., whether it reports an actual failure to forward a packet) cannot be verified though. As a result, a malicious node lying on a route can mislead the source by corrupting error messages generated by another node, or by masking a dropped packet as a link failure. However, this allows it to harm only the route it belongs to, something that was possible in the first place, if it simply dropped or corrupted in-transit data packets.

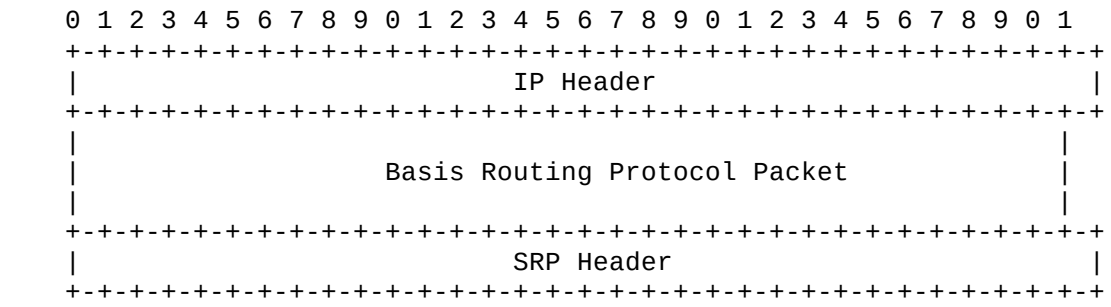
ROUTE_ERROR messages do not include a MAC if the reporting intermediate node does not have a security association with the source node. This allows an adversary that can spoof a data link address and lies within hop of an end-to-end data flow (route) to inject a ROUTE_ERROR. This would be possible if it impersonated a node that is part of the route. Although the NLP of the victim would issue a notification, the forged ROUTE_ERROR would be in-transit towards the source.

Consequently, ROUTE_ERROR messages can be used in the following cases: (i) an end-to-end secure mechanism is present and thus the source node can infer the status of the utilized route(s) the intermediate issuing node has a secure association with the source node, (ii). In case (i), the ROUTE_ERROR packets should be used only in a complementary manner. For example, our Secure Message Transmission (SMT) protocol [20], which provides a robust, secure, end-to-end feedback mechanism, can utilize unauthenticated ROUTE_ERROR messages to update the ‘rating’ of the utilized route(s) only when the end-to-end feedback reports a failed transmission. In case (ii), an intermediate node, which is for example member of the same group as the source of the broken route, SHOULD use the group key to generate a ROUTE_ERROR MAC that covers the entire packet and its IP source route.

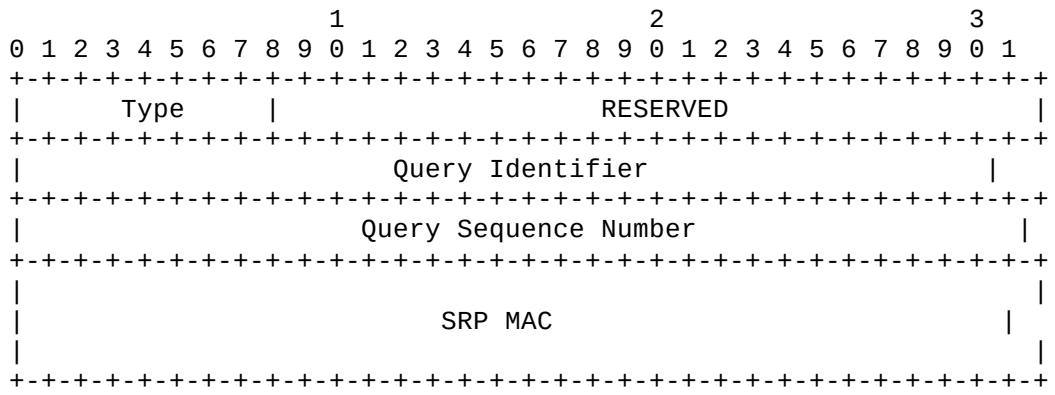
4. Implementation Details

4.1 Packet Formats

A. Position of the SRP header: the SRP header (shown in detail in 4.1.B) is appended to the basis routing protocol header.



B. The SRP Header



Field Description:

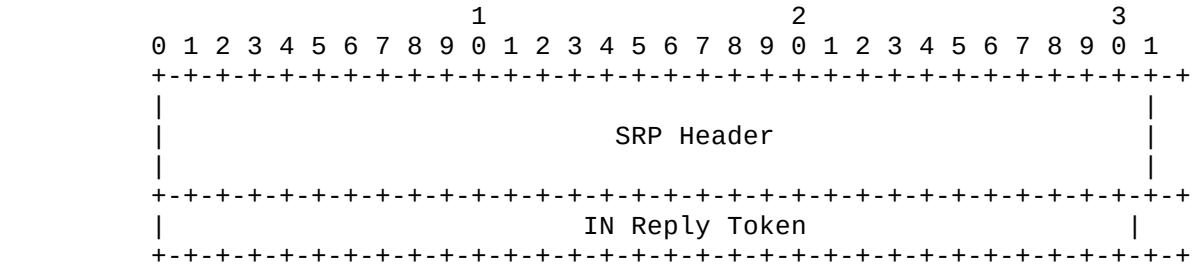
* Type (char) (8 bits)
  Identifies the type of SRP packet. The current version of SRP contains two packet types:
  ROUTE_REQUEST:
  Request for a route to the Query Destination.
  ROUTE_REPLY:
  Response to a ROUTE_REQUEST packet, issued by the destination (or the node that has an active route to the Query Destination, if intermediate node replies are allowed), and sent back to the Query Source.

* Query Identifier (unsigned int) (32 bits)
  A random number generated by the source which, along with the Query Source Address, is used by the intermediate nodes to uniquely identify a route request.

* Query Sequence Number (unsigned int) (32 bits)
  A monotonically increasing sequence number associated with each destination that a source node communicates with.

* SRP MAC (unsigned int) (96 bits)
  The Message Authentication Code which is the output of a hash function using the secret key K(S,T) shared by the two communicating nodes.

C. Extended SRP Header



Field Description:

* SRP Header
  As described in 4.1.B.

* IN Reply Token
  The Intermediate Node Reply Token is a Message Authenticaion Code (MAC) of the route query/reply calculated using the secret group key, K(G).

4.2 Data Structures

A. Query Table

Source Address (node_id)	Destination Address (node_id)	Q_ID (unsigned int)	FORWARD_LIST (node_id *)

5. Discussion

SRP can guarantee correct connectivity information if a set of malicious nodes mount attacks against the protocol concurrently, but are not capable of colluding within one step of the protocol execution; that is, within the period of broadcasting one query and reception of the corresponding reply. If this is possible, then, the malicious nodes can tunnel the control traffic between themselves and cause the querying node to accept partially incorrect link-level connectivity information [1]. However, this vulnerability is not pertinent to SRP; such routing information would be distinguished even under much stronger or full trust assumptions. In any case, SRP prevents the two colluding adversaries to manipulate the prefix and suffix of the discovered route; they can only arbitrarily define the portion between them, without any ‘guarantee’ that such a route request will result in a discovered and partially incorrect route (which, nonetheless, provides for the fact that there is end-to-end connectivity).

6. References

[1] P. Papadimitratos and Z.J. Haas, "Secure Routing for Mobile Ad Hoc Networks," SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002), San Antonio, TX, January 27-31, 2002.

[2] P. Papadimitratos, "Securing the Internet Routing Infrastructure,” IEEE Communications Magazine, October 2002.

[3] L. Lamport, R. Shostak, M. Pease, "The Byzantine Generals Problem," ACM Trans. Program. Languages, Vol. 4, no. 3, pp. 382-401, July 1982.

[4] L. Zhou and Z.J. Haas, "Securing Ad Hoc Networks," IEEE Network Magazine, vol. 13, no.6, November/December 1999.

[5] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues for Ad Hoc Wireless Networks, "Security Protocols, 7 th International Workshop, LNCS, Springer-Verlag, 1999.

[6] IEEE Std. 802.11, "Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[7] R. Zucceratto, and C. Adams, "Using Elliptic Curve Diffie-Hellman in the SPKM GSS-API," Internet Draft, IETF, Aug. 1999.

[8] D. B. Johnson et al, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," Internet Draft, IETF MANET Working Group, March 2nd, 2001.

[9] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," RFC 2104, February 1997.

[10] M. Hattig, Editor, “Zero-conf IP Host Requirements,” draft-ietf-zeroconf-reqts-09.txt, IETF MANET Working Group, Aug. 31st, 2001.

[11] Alfred Menezes, Paul van Oorschot and Scott Vanstone, "Handbook of Applied Cryptography," CRC Press, October 1996 5th reprinting, Aug. 2001.

[12] W. Diffie, M.E. Hellman, "New directions in cryptography," IEEE Transactions in Information Theory, 1976.

[13] Z.J. Haas, M.R. Pearlman, P. Samar, "The Interzone Routing Protocol (IERP) for Ad Hoc Networks," IETF MANET Working Group, July, 2002.

[14] Z.J. Haas, M.R. Pearlman, P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks," IETF MANET Working Group, July, 2002.

[15] C.K. Toh, "Associativity-Based Routing for Ad-Hoc Mobile Networks," Wireless Personal Communications, Vol. 4, No. 2, pp. 1-36, Mar. 1997.

[16] NIST, Fed. Inf. Proc. Standards, "Secure Hash Standard," Pub. 180, May 1993.

[17] R. Rivest, "The MD5 Message-Digest Algorithm," RFC 1321, April 1992.

[18] W. Stevens, “Unix Network Programming,” Prentice-Hall.

[19] I. Avramopoulos, personal communication.

[20] P. Papadimitratos and Z.J. Haas, “Secure Message Transmission for Mobile Ad Hoc Networks,” submitted for publication.