Secure Routing & Data Transmission in Mobile Ad Hoc Networks

Nonika Sharma, Priyanka Sahni
Information Technology, Dronacharya College Of Engg, Gurgaon,Haryana

***Abstract*- In this paper we present an identity (ID) based protocol that secures AODV and TCP so that it can be use in dynamic and attack prone environment of mobile ad hoc networks. The proposed protocol protects AODV sequential aggregates signature(SAS)based on (RAS).It also generates a session key for each pair of source-destination nodes of a MANET for securing the end-to-end transmitted data. Here each node has an ID which is evaluated from its public key and the messages that are sent are authenticated with a signature/MAC. The proposed scheme does not allow anode to change ID throughout the network lifetime. Thus it makes the network secure against attack the target AODV and TCP in MANET. We present performance analysis to validate our claim.**

***Index Terms*- Fault tolerance, mobile ad hoc network (MANET) security, multipath routing, network security, secure data transmission, secure message transmission, secure routing.**

I. INTRODUCTION

The emerging technology of *mobile ad hoc networking* (MANET) is based on wireless multihop architecture without fixed infrastructure and prior configuration of the network nodes. The salient features of this new networking paradigm include: 1) collaborative support of basic networking functions, such as routing and data transmission; 2) lack of administrative boundaries of the network nodes; 3) absence of a central entity in the network; and 4) transient, in general, associations of the network nodes. As a result, a node cannot make any assumption about the trustworthiness of its peers, which assist the node with its communication and, in general,does not possess their credentials site for reliable and quality-of-service (QoS) communication in adversarial environments. The challenge lies in securing communication and maintaining connectivity in the presence of adversaries, across an unknown, frequently changing multihop wireless network topology. To address this complex problem and provide comprehensive security, both phases of the communication, the route discovery and the data transmission, must be safeguarded.

Recently, a number of works proposed secure routing mechanisms to defend against a range of attacks under different assumptions and system requirements. However, secure routing protocols alone, which ensure the correctness of the route discovery, cannot guarantee secure and undisrupted delivery of data. In other words, a correct, up-to-date route cannot be considered automatically free of adversaries. An intelligent adversary can, for example, follow the rules of the route discovery, place itself on a route, and later start redirecting traffic, dropping, or forging and injecting data packets. Clearly, an adversary can hide its malicious behavior for a long period of time and strike at the least expected time. Thus, it is impossible to discover such an adversary prior to its attack.

MANET routing, as well as secure routing protocols assume mechanisms, such as reliable data link layer and route maintenance, which were not designed for and cannot cope with malicious disruptions of the data transmission. Reliable transport protocols cannot address the problem either: an attacker can forge, for example, transmission control protocol (TCP) acknowledgment, while dropping data packets, misleading two communicating nodes that the data flow is undisrupted. End-to-end security such as the IP-Security (IPSec) *authentication header* (AH) protocol can prevent adversaries from forging or corrupting data and feedback. But IPsec does not allow the sender to detect loss of data and, thus, take any corrective action. Nor the combination of security services and reliable transport .e.g., *stream control transmission protocol* (SCTP) provides an effective solution: a communication failure can be detected, but the same, structurally intact yet compromised path will be repeatedly utilized, because the transport layer protocol cannot influence

the choice of the route in the network. Finally, multipath transmissions can protect against failures. However, "blind" redundant transmissions alone can be highly inefficient without a robust mechanism to detect transmission failures and adapt to the network loss conditions.

Our contribution is a novel, general solution, tailored to the MANET requirements, to effectively and efficiently secure the data transmission phase: the *secure message transmission* (SMT) and *secure single-path* (SSP) protocols. We emphasize that the goal of SMT and SSP is not to securely discover routes in the network—they assume that secure discovery of routes has been already performed, although routes may not be free of adversaries. Then, the goal of SMT and SSP, whose basic ideas we presented is to secure the data transmission: SMT and SSP operate without restrictive assumptions on the network trust and security associations, promptly detect and avoid nonoperational or compromised routes, tolerate loss of data and control traffic, and adapt their operation to the network conditions. Their main difference is that SMT utilizes multiple paths simultaneously, in contrast to the single-path operation of SSP.

II. NETWORK AND SECURITY MODEL

We define a network *node* as a process with: 1) a unique identity V;2) a public/private key pair Ev,Dv;3) a module implementing the networking protocols, e.g., routing, data transmission; and 4) a module providing communication across a wireless network interface. The combination of an *Internet protocol* (IP) address and a public key can uniquely identify a node.

III. OVERVIEW OF SMT

SMT requires a security association (SA) only between the two end communicating nodes – the source and the destination. Since a pair of nodes chooses to employ a secure communication scheme, their ability to authenticate each other is indispensable. The trust relationship can be instantiated, for example, by the knowledge of the public key of the other communicating end. However, none of the end nodes needs to be securely associated with any of the remaining network nodes. As a result, SMT does not require cryptographic operations at these intermediate nodes. With SMT, at any particular time, the two communicating end nodes make use of a set of diverse, preferably node-disjoint paths that are deemed valid at that time. We refer to such a set of paths as the Active Path Set (APS). The source first invokes the underlying route discovery protocol, updates its network topology view, and then determines the initial APS for communication with the specific destination. With a set of routes at hand, the source disperses each outgoing message into a number of pieces. At the source, the dispersion, based on the algorithm in [3], introduces redundancy and encodes the outgoing messages, as described in Section 3.2. At the destination, a dispersed message is successfully reconstructed, provided that sufficiently many pieces are received. In other words, the message dispersion ensures successful reception even if a fraction of the message pieces is lost or corrupted, either due to the existence of malicious nodes, or due to the unavailability of routes (e.g., breakage of a route as a result of nodes' mobility). Each dispersed piece is transmitted across a different route and carries a Message Authentication Code (MAC) [4], so that the destination can verify its integrity and the authenticity of its origin. The destination validates the incoming pieces and acknowledges the successfully received ones through a feedback back to the source. The feedback mechanism is also secure and fault tolerant: it is cryptographically protected and dispersed as well. This way, the source receives authentic feedback that explicitly specifies the pieces that were received by the destination. A successfully received piece implies that the corresponding route is operational,
5 while a failure is a strong indication that the route is either broken or compromised.

While transmitting across the APS, the source updates the rating of the APS paths. For each successful or failed piece, the rating of the corresponding path is increased or decreased respectively, as we explain in Section 3.3. A path is discarded once it is deemed failed
and a precaution is taken not to use the same path, if it is discovered again within sometime after it has been discarded. Whil continuously assessing the quality of the utilized paths, th protocol adapts its operation according to the feedback it receives from

the trusted destination. Based on its interaction with the network, the protocol adjusts its configuration to remain effective in highly adverse environments and efficient in relatively benign conditions.

To secure the data transmission phase, we propose and evaluate the Secure Message Transmission (SMT) protocol, an end-to-end secure data forwarding protocol tailored to the MANET communication requirements. The SMT protocol safeguards pairwise communication across an unknown frequently changing network, possibly in the presence of adversaries that may exhibit arbitrary behavior. It combines four elements: end-to-end secure and robust feedback mechanism, dispersion of the transmitted data, simultaneous usage of multiple paths, and adaptation to the network changing conditions. SMT detects and tolerates compromised transmissions, while adapting its operation to provide secure data forwarding with low delays. We underline that the goal of SMT is not to securely discover routes in the network – the security of this phase should be achieved by one of the protocols proposed in the literature [1,2,5,23-25]. 1 The goal of SMT is to ensure secure data forwarding, after the discovery of routes between the source and the destination has been already performed. In other words, SMT assumes that there is a protocol that discovers routes in the ad hoc network, although such discovered routes may not be free of malicious nodes. 2 Then, the goal of SMT is to ensure routing over such routes, despite of the presence of such adversaries. In addition to SMT, we present and evaluate here the Secure Single Path (SSP) protocol, an end-to-end secure data forwarding protocol that utilizes a single route. Unlike SMT, SSP does not incur multi-path transmission overhead. Thus, it does not require that the underlying routing protocol discover multiple routes either. As a result, SSP imposes less routing overhead per discovery than SMT. Overall, we examine SSP and compare it to SMT as an alternative, lower cost, more flexible protocol to secure the dataforwarding phase. Our results show that SMT outperforms SSP consistently over a wide range of experiments. The advantages of SMT over SSP become more pronounced in highly adverse environments: SMT delivers up to 22% more data packets than SSP, and achieves up to 94% lower delays than SSP. It is also very interesting that SMT imposes up to 68% less routing overhead than SSP, although overhead was expected to be lower for SSP. In contrast, SSP provides only up to 48% lower transmission overhead than SMT.

We especially emphasize the low-delay characteristic of SMT, as we believe that one of the main applications of SMT is in support of QoS for real-time traffic. In the rest of the paper, we first provide an overview of the SMT protocol and present its operation. Then, in Section 4, we outline the operation of SSP and evaluate the performance of the two protocols. Related work is discussed next, followed by a discussion and description of future work in Section 6, before our conclusion.. If a sufficient number of pieces are received at the destination, the destination proceeds to reconstruct the message. Otherwise, if a dispersed message cannot be reconstructed at the destination, it awaits the missing packets that are retransmitted by the source. The number of re-transmissions is limited to Retrymax per serviced message. An illustrative example of a single message transmission is shown in Fig. 1. The sender disperses the encoded message into four packets, so that any three out of the four packets are sufficient for successful reconstruction of the original message. The four packets are routed over four disjoint paths and two of them arrive intact at the receiver. The remaining two packets are compromised by malicious nodes lying on the corresponding paths; for example, one packet is dropped, and one (dashed arrow) is modified.

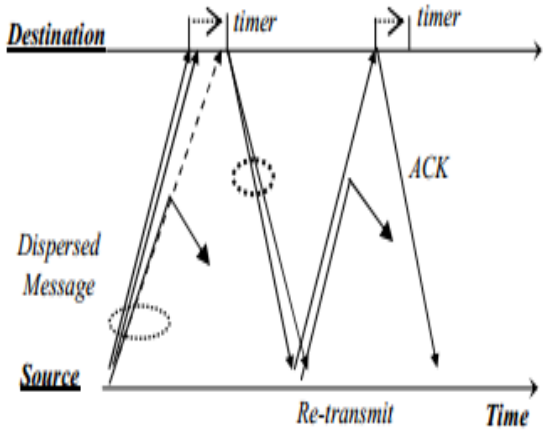

Figure 1. Simple example of the SMT protocol.

IV. MESSAGE DISPERSION AND TRANSMISSION

The information dispersal scheme is based on Rabin's algorithm, which acts in essence as an erasure code: it adds limited redundancy to the data to allow recovery from a number of faults. The message and the redundancy are divided into a number of pieces, so that even a partial reception can lead to the successful reconstruction of the message at the receiver. In principle, the encoding (and dispersion) allows the reconstruction of the original message with successful reception of any M out of N transmitted pieces. The ratio r = N/M is termed the redundancy factor. Messages, i.e., raw data, can be viewed as a stream of integers, or m-bit characters, so that each integer is in the [0…2-1] range. It suffices to select a prime number p > 2
m-1, so that all encoding and decoding operations are performed in a finite field mod p. Initially,
N random M-vectors, organized as rows {ai} of matrix A, are selected, with any M of them linearly independent. These ai vectors can be constructed by selecting N different elements ui of the finite field and set ai=[1,ui,…,ui M-1], 1≤ i≤ N, and N<p. The vectors of matrix A should be selected from a pre-computed set used by both ends, which we assume are agreed upon as part of the SA establishment process. The encoding of a message first segments the original message of
length FS into L sequences of characters, each of length M, with padding if necessary. The segments of the original message are denoted by s1, s2, …, sL and they are arranged as columns of an Mby-L array B. Then, each piece wi of the dispersed message is created as a character sequence of length L: to do so, the original message segments are multiplied by the corresponding random vector ai, and the resultant piece is wi=[ais1, ais2, …, aisL]. Upon reception of any M pieces, the original message can be reconstructed. Let v1, v2,…, vM denote the M pieces used for reconstruction, which are in fact a subset of the N transmitted

V. PERFORMANCE EVALUATION

Our experiments verify that the proposed protocol can, indeed, successfully cope with a high number of adversaries, while operating only in an end-to-end manner. SMT can deliver successfully more than twice the number of packets delivered by a protocol that secures only the route discovery phase but not the dataforwarding phase. Moreover, we find that SMT is successful in delivering data with low end-to-end delay, low routing overhead, and limited transmission overhead, when compared to SSP. The Secure Single Path (SSP) protocol is the limiting case of SMT without the dispersion of outgoing messages and the use of a single path for each message transmission. SSP is equipped with the same end-to-end feedback and the fault detection mechanisms as SMT. SSP also re-transmits each failed message Retrymax times, provides data integrity, authenticity, and replay protection as SMT does, and selects the shortest path in hops. SSP determines, utilizes, and maintains a single path only. Once the utilized path is deemed failed, a new route discovery may be needed in order to determine a new route.
We evaluate here three protocols: (i) a single-path data forwarding protocol that does not employ any security mechanism to protect data transmissions, which we term the Non-Secure Single Path (NSP) protocol, (ii) the SSP protocol, and (iii) the SMT protocol. In all cases, we assume that the route discovery is secured, that is, the correctness of the discovered connectivity information is guaranteed. 12 Here, the secure discovery of one or more routes is 11 If the content of the packets can be analyzed, the attack could be selective, targeting packets of high importance. The selection of the packets to corrupt could depend on the knowledge of the employed protocols and the supported applications or could be purely subjective. For example, the loss of the last message of a multi-round interactive protocol can have a severe impact. 12 But, again, this does not imply paths free of malicious nodes.

VI. DISCUSSION AND FUTURE WORK

In this work, we showed how the data-forwarding phase can be secured by a protocol that operates solely in an end-to-end manner, without any further assumptions on the network trust and behavior of the adversaries. In fact, SMT can counter any attacker pattern, either persistent or intermittent, by promptly detecting nonoperational or compromised routes. Moreover, SMT bounds the loss of data incurred by an intelligent adversary that avoids detection through manipulation of the path rating scheme. At the same time, SMT provides robustness to benign network

faults as well, whether transient or not. The resilience to transient faults is very important, as it avoids discarding routes that are operational, 14 thus avoiding unnecessary overhead. Furthermore, resilience to benign faults, along with malicious ones, is important, since in MANET they may be frequent and in practice indistinguishable from forms of denialof-service attacks. Fault tolerance is dependent on the ability of the protocol to determine and utilize alternative, new routes when it detects nonoperational ones. The multiplicity of routes that are, in general, expected to be available in MANET multi-hop topologies can be clearly beneficial. The availability or timely determination of such redundant routes may be the single most important factor for successful transmission across an adverse network. A rich APS, or many alternative routes, can be available only at the expense of routing overhead. This is generally true for any underlying routing protocol, even though the exact amount and type of routing overhead depends on the employed routing protocol. Increasing the size of the APS will most probably increase the routing overhead, which, in the case of reactive routing protocols, may result from more frequent
route requests and additional replies, or, in the case of proactive protocols, more frequent link state updates. However, by trading off higher routing overhead, increased reliability (that is, higher fraction of delivered messages) and lower delays can be achieved. In fact, the number of available diverse routes appears to control the trade-off between the delay, the routing and the transmission overhead, and the fraction of delivered messages. For example, the larger the size of the utilized APS, the more probable the successful reconstruction of the dispersed message will be and, consequently, the fewer the data re-transmissions and, thus, the lower the message
delay. The protocol adapts to either reduce the overhead or increase its fault tolerance, by selecting for each message the number of paths, among those available, and the redundancy factor. It starts with selecting an APS of K shortest (in terms of hops) paths [21].

Without having the opportunity to "probe" the paths and assuming that initially all nodes are equally probable to be malicious, selecting the shortest paths is equivalent to the selection of the most secure paths. The source maintains an estimate, pi, of the probability that each APS path is operational. For each combination of the number of paths, m, and the feasible values of r, the probability that a transmission is successful is calculated with the estimated values for pi–s in hand. The source selects m and r that yield a probability of successful delivery equal or as close as possible to the required probability of successful message delivery, PGOAL, (determined, for example, by the application layer). The reader is referred to [28] for additional discussion and implementation details. An open issue of interest is how to obtain estimates or predictions
of the probability that a route will be operational. The complexity of such a task is increased, because of the numerous factors that affect the condition of the utilized routes. Mobility, congestion, transmission impairments, and an arbitrary, possibly intermittent and changing over time attack pattern, have to be taken into consideration. Through its interaction with the network and the feedback it obtains from the trusted destination, each node can gradually 'construct' such estimates. Clearly, the network conditions and characteristics can change over time. More simply, parameters such as the network connectivity, density, or the number of attackers present can differ according to the nodes' neighborhood. In any case, a feasible estimation method would be able only to continuously track such changes and to provide rough estimates.

A plausible approach to obtain the probabilities of operational routes would be to collect statistics on the lifetimes of all the utilized routes. It would be helpful to categorize routes according to attributes such as the length or whether the route includes any additional trusted nodes, other than the destination. Moreover, it would be more meaningful to update such measurements by assigning a lower weight to earlier observations in order to account
for the network dynamics. For example, a node could quantize path lifetimes and retain measurements and estimates for a set of intervals. Then, if a newly determined path of length i has been operational for a period t in the [tx,tx+1] interval, the node utilizes the estimate of the probability that such a path will survive for a period t' > t, with t' in the [tx+1,tx+2] interval. The investigation and evaluation of such mechanisms are left as future work. Finally, we note that, despite the use of re-transmissions, SMT does not assume the role of a transport layer protocol - it operates at the network layer to secure the data

forwarding and improve significantly the reliability of message delivery. However, SMT provides security and protects from frequent disruptions at the expense of increased traffic at the network, especially when data loss is detected. If there is not enough capacity in the network (at the link and at the network layers) to accommodate both the data flows and the SMT's overhead, the upper layer data rate could be decreased, for example, by the congestion control mechanism of the transport layer protocol.

VII. CONCLUSIONS

In this paper, we have presented the SMT protocol to secure the data forwarding operation for MANET routing protocols. Our protocol takes advantage of topological and transmission
redundancies and utilizes feedback, exchanged only between the two communicating end-nodes. This way, SMT remains effective even under highly adverse conditions. Moreover, features such as low-cost encoding and validation mechanisms, and partial retransmissions render the scheme efficient. By relying solely on the end-to-end security associations, SMT can secure effectively the data transmission without prior knowledge of the network trust model or the degree of trustworthiness of the intermediate nodes. Our performance evaluation confirms that SMT can naturally complement any protocol that secures the route discovery and can shield the network operation by delivering up to 250% more packets despite the presence of substantial fraction of nodes as attackers. We also confirmed that SMT outperforms SSP, a single-path secure data transmission protocol equipped with the SMT's mechanisms. The end-to-end delays achieved by SMT are up to 94% lower than the delays of SSP. Yet, SMT delivers up to 22% more messages.

And it does so with 68% lower routing overhead and only with up to 48% data and feedback transmission overhead. In conclusion, SMT's low overhead and its efficient and effective operation render SMT applicable to a wide range of MANET instances. The highly successful delivery of messages, in spite of the presence of adversaries and, most importantly, the low end to-end delay clue on the ability of the protocol to support QoS for real-time traffic.

REFERENCES

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

[2] P. Papadimitratos, Z.J. Haas, and P. Samar, "The Secure Routing Protocol (SRP) for Ad Hoc Networks," Internet Draft, draft-papadimitratos-secure-routing-protocol-00.txt, Dec. 2002.

[3] M.O. Rabin, "Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance," Journal of ACM, Vol. 36, No. 2, pp. 335-348, Apr. 1989.

[4] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: KeyedHashing for Message Authentication," RFC 2104, Feb. 1997.

[5] P. Papadimitratos and Z.J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," in Proceedings of the IEEE CS Workshop on Security and Assurance in Ad hoc Networks, in
conjunction with the 2003 International Symposium on Applications and the Internet, Orlando, FL, Jan. 2003.

[6] A. Tsirigos and Z.J. Haas, "Multipath Routing in the Presence of Frequent Topological Changes," IEEE Comm. Magazine, pp. 132-138, Nov. 2001.

[7] J. Broch, D.A. Maltz, D.B. Johnson, Y-C. Hu, J. Jetcheva,
"A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," in proceedings of the 4th International Conference on Mobile Computing (Mobicom'98), 1998.