Self-Organized Trust-Based Public-Key Security Management for Mobile Ad Hoc Networks

K.Nikhila Reddy
PG Student
TRR Engineering College, JNTU

N.Naresh Reddy
Asst.Professor
Gopal Reddy Engineering College

Dr.P.Raja Prakash Rao
Professor & Head of CSE
TRR Engineering College, JNTU

Abstract

MANETs (Mobile Ad hoc networks) are unplanned, self-organizing networks composed of mobile nodes that utilize mesh networking principles for inter-connectivity. MANETs have several advantages compared to traditional wireless networks. These include ease of deployment, speed of deployment and decreased dependency on a fixed infrastructure. There have been many studies done in this area to improve the quality and efficiency of the routing protocols in MANETs. However unique characteristics of MANETs topology such as open peer-to-peer architecture, dynamic network topology, shared wireless medium and limited resource (battery, memory and computation power) pose a number of non-trivial challenges to security design. The fixed infrastructure less environment makes the transactions less secure. Existing traditional routing algorithms in MANETs do not work with cryptographic techniques.

In this paper, we propose Cryptography procedures to make secure transactions. Security is paramount in Mobile Ad-hoc Networks (MANET) as they are not conducive to centralized trusted authorities. Several solutions have been proposed MANET in the areas of key management, secure routing, nodal cooperation, and trust management. In this work, we are focusing on the evaluation of trust evidence in Ad Hoc Networks.

Index Terms --- Mobile Ad Hoc Network, Trust-Based Security, Ad Hoc Routing, Attacks in MANETs, Public Key Distribution, Cryptography.

I. INTRODUCTION

BY definition, a mobile ad hoc network [1], [2] does not rely on any fixed infrastructure; instead, all networking functions (e.g., routing, mobility management, etc.) are performed by the nodes themselves in a self organizing manner. For this reason, securing mobile ad hoc networks is challenging and, as we show in this paper, in some applications this requires a shift in paradigms with respect to the traditional security solutions for wireless networks. Meanwhile, we still rely on traditional cryptographic primitives.

In our view, there are two extreme ways to introduce security in mobile ad hoc networks:

1) Through a single authority domain, where certificates and/or keys are issued by a single authority, typically, in the system setup phase or
2) through full self-organization, where security does not rely on any trusted authority or fixed server, not even in the initialization phase. Ad hoc routing protocols must be integrated into authentication architectures, such as public key infrastructure (PKI) and certificate authority (CA), to achieve the security requirements including confidentiality, integrity, authentication, and non-repudiation services. Thus in our project, we take the second approach and we propose a self-organizing public-key management system that allows users to create, store, distribute, and revoke their public keys without the help of any trusted authority or fixed server. Moreover, in our solution, we do not assign specific missions to a subset of nodes (i.e., all the nodes have the same role). Our main motivation for taking this approach comes from the self-organized nature of mobile ad hoc networks and from the need to allow users to fully control the security settings of the system. As such, our approach is developed mainly for "open" networks, in which users can join and leave the network without any centralized control.

In this paper, we make the following contributions to the area of secure routing protocols for ad hoc networks. First, how to detect and defend internal attacks against routing protocols has been a particularly challenging problem. The problem has often been avoided by most secure routing protocols by assuming that the nodes should be trusted once authenticated. This is, unfortunately, not the case for real-world environments. Second, what kind of authentication and key management schemes are needed to dynamically maintain a trustworthy topology and defend against malicious attacks? The security measures in mobile telecommunication networks can rely on a CA or ID-based cryptosystem. However, a MANET cannot use such a CA server. Thus, The main problem of any public-key based security system is to make each user's public key available to others in such a way that its authenticity is verifiable. Third, the existing practice in developing secure routing protocols is by first establishing a PKI and then using cryptographic primitives to protect the messages exchanged in the routing protocols. The security and routing mechanisms are separately designed to meet the conflicting requirements: security requires using intensive computations, whereas routing needs to be efficient to properly scale. Thus, the resulting protocols may be secure but not feasible or vice versa. This paper proposes a novel attack detection and defense algorithm to solve the preceding problems for MANETs. Fourth, As secure routing protocols are not designed to guarantee the availability of network, they are extremely vulnerable to attacks such as flooding and packet drop attacks. These attacks completely disrupt the functioning of network. Although solutions have been proposed [3, 4] to induce cooperation among nodes, they fail to counteract flooding attacks. The reason rests on the fact that cooperation models fail to consider the behavioral patterns of nodes, and hence overlook to measure the trustworthiness for nodes. Recently, few reputation and trust models have been proposed [5, 6] to evaluate the trustworthiness for intermediate nodes. However, these models introduce additional issues and modify the basic routing operations in order to collect evidence of trustworthiness for intermediate nodes.

The rest of this paper is organized as follows. In Section II, we report on existing approaches to define the general security architecture for MANETs. We then provide the general routing protocols in ad hoc networks in Section III. Security being the core issue in MANETs we present the types of major attacks in Section IV. This is followed by Cryptographic solution to security breaches in MANETs in Section V. We then demonstrate how trust is evaluated among the nodes in Section VI. The last section concludes the paper.

II. RELATED WORK

In works on security for MANETs Ariadne [7] employs broadcast and hop-by-hop authentication, while Secure Routing Protocol (SRP) [8] performs end-to-end authentication through symmetric key based mechanism. These secure routing protocols are primarily designed to discover secure paths, and therefore they fail to defend against both flooding and packet drop attacks. Zhou and Hass [9] introduce two types of nodes known as server and combiner apart from the normal nodes (called as client nodes) to play the role of Certificate Authority (CA). They deploy threshold based cryptography to establish the services of CA. In [10], Luo et al. replaced the abovementioned specialized server nodes by distributing the capability of CA to all nodes. Distributed Key Pre-distribution Scheme (DKPS) [11] is a fully distributed and self-organized key pre-distribution which does not rely on any infrastructure support. Nevertheless, these approaches are prone to refresh keys with malicious and compromised nodes.

Nuglets [3] enforces nodes to cooperate by using virtual currencies. In spite of efficiency, the usage of tamper-resistant hardware makes Nuglets unattractive. Alternatively, Sprite [4] uses incentives to motivate cooperation among selfish nodes. Sprite is non-generic as it relies on a central authority to manage incentives and also fails to address malicious nodes. In [5], Liu and Yang collect reputations from recommenders and combine them to update the reputation for recommended node. The main drawback of the model is that the malicious nodes are assumed to recommend truthfully irrespective of their misbehaviors. Yan Lindsay et al. [6] proposed a trust model based on information theory for improving the security of ad hoc routing protocols. Similar to other models, they monitor other nodes and exchange recommendations with other nodes in a distributed manner for establishing trust relationships.

III. ROUTING IN MANETS

In mobile ad-hoc networks where there is no infrastructure support as is the case with wireless networks, and since a destination node might be out of range of a source node transmitting packets; a routing procedure is always needed to find a path so as to forward the packets appropriately between the source and the destination. Within a cell, a base station can reach all mobile nodes without routing via broadcast in common wireless networks. In the case of ad-hoc networks, each node must be able to forward data for other nodes. This creates additional problems along with the problems of dynamic topology which is unpredictable connectivity changes [12].

Asymmetric links: Most of the wired networks rely on the symmetric links which are always fixed. But this is not a case with ad-hoc networks as the nodes are mobile and constantly changing their position within network. For example consider a MANET where node B sends a signal to node A but this does not tell anything about the quality of the connection in the reverse direction [13].

– Routing Overhead: In wireless ad hoc networks, nodes often change their location within network. So, some stale routes are generated in the routing table which leads to unnecessary routing overhead.

– Interference: This is the major problem with mobile ad-hoc networks as links come and go depending on the transmission characteristics, one transmission might interfere with another one and node might overhear transmissions of other nodes and can corrupt the total transmission.

– Dynamic Topology: This is also the major problem with ad-hoc routing since the topology is not constant. The mobile node might move or medium characteristics might change. In ad-hoc networks, routing tables must somehow reflect these changes in topology and routing algorithms have to be adapted. For example in a fixed network routing table updating takes place for every 30sec [13]. This updating frequency might be very low for ad-hoc networks.

IV COMMON ATTACK SCENARIOS

There are six main properties that any secure networking system should be able to provide: Secrecy, authenticity, integrity, availability, non-repudiation, and access control. It is a breach of security if one or more of these security objectives are contravened by any attack on a computer system. Some of the most common attacks that occur on a distributed computer system [21] are as given below:

• Denial of Service: This takes place when there is non-availability of a network service owing to excess load or breakdown.

• Information theft: This occurs when interpretation of data is through an unauthorized instance.

• Intrusion: This happens when unauthorized person gains admittance to several restricted services

• Tampering: This ensues when data is distorted by an unauthorized person.

Attacks and security goals that an ad hoc network encounters is similar to that of other networks. It becomes easy to gain access to data or to lose the stored (e.g. passwords, cryptographic keys, etc.) data on a node because the substantial network contributors are mobile devices. Based on these factors, it is all the more important that in an ad hoc network the overall security should not depend on any one factor. One of the most popular means of communication in mobile networks is radio transmission. Eavesdropping on a node is much easier vis-à-vis wired networks. Sometimes intermediate nodes maybe disguised eavesdropper and not linked to some trusted infrastructure. Therefore, end-to-end encryption becomes a vital issue that has to be dealt with in all cases. This is usually the case because all the nodes of an Ad hoc network work together to make the discovery of network typology and forward packets easy to overcome. These nodes can also produce stale or wrong routes, black holes or routing loops. In addition, there is a strong momentum available for non-participation in the routing system of an Ad hoc network. Selfish nodes often want to hoard resources for their own use due to consumption of a node's battery power, CPU time, and bandwidth in both the routing system and the forwarding of foreign packets which are limited in mobile devices.

V. DYNAMIC KEY MANAGEMENT SCHEME

In the network layer, the most possible attacks are data and routing information tampering. The majority of external attacks against routing protocols can be prevented by simple link layer encryption and authentication. We propose to have every node share a unique symmetric key with the source if it needs to transmit data.

A. Dynamic Key Management Scheme There are two basic key management approaches, i.e., public and secret key-based schemes. The public key-based scheme uses a pair of public/private keys and an asymmetric algorithm such as RSA to establish session keys and authenticate nodes. In the latter scheme, a secret key is a symmetric key shared by two nodes, which is used to verify the data integrity.

There are several methods to set up the shared keys: 1)Bootstrap the shared keys from a PKI, which might be a strong assumption for MANETs; 2) use a key distribution center, which has a shared key with each node, to build up a shared key between two nodes by using the Kerberos protocol; or 3) embed the shared keys in each node during its initialization before deployment. In this paper, we assume that each node has a unique ID or address and an initial pair of public/private keys, which can be embedded into each node at the initialization of the network, or created by a self organized public key management system.

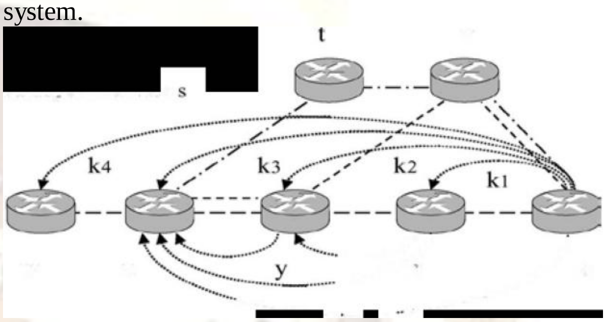

Fig 1. Demonstration of message and route redundancy. Multiple secret keys are shared between a source and the intermediate nodes and the destination node. Multiple copies of a message are received at a destination node via different routes.

We first define a network, as shown in Fig. 1, and then describe a framework of dynamic key management. Let G = (V ;E) be a network whose vertices in V are nodes and whose edges in E are direct wireless links among nodes. We define for each node x the set $N_1(x)$, which contains the vertices in the network G that are hop-l or direct neighbors of x, i.e

$N_1(x)$= (y : (x;y) £ E and y ≠ x)   (1)

Similarly, we define the hop-2 neighbors of a node as follows. For each node x, $N_2(x)$ contains the vertices in the network G that are hop-2 neighbors of x, which include neither vertices in $N_1(x)$ nor x itself, i.e.,

**$N_2(x)$= { z: (y;z) £ E and y € $N_1(X)$, z ≠ x }
(2)**

Similarly, we can define the hop-n neighbors of x [Nn(x)] in terms of Nn-$_1$(x) if the flooding path from the source to destination has n links. As in the existing secure routing protocols, the initial trust among the nodes is built into the network by using some external mechanisms. After that, unlike the existing secure routing protocols, our framework allows a node to build up its trust on its neighboring nodes based on its observations of their behaviors. Here, important behavior is whether a node correctly routes and forwards a message to its neighbors. Initially, a node x has a public key $K_{x,pub}$ that is distributed to $N_1(x)$ by using PKI or CA. Similarly, a node y has public key

$K_{y,pub}$ distributed to $N_1(y)$. Thus, for example, if y £ $N_1(x)$ and x £ $N_1(y)$, i.e., x and yare hop-l neighbors, then x can authenticate y by issuing a certificate (which is a proof of y' sID and public key with x's signature) that is signed by x with x's private key. Those who hold x's public key can now read the certificate and trust the binding of y and its public key. Based on the available certificate and key information, two hop-l neighboring nodes can easily establish a secret key between them by using methods such as a three-way handshake.

Key Distribution and Node Authentication

We define the notations as follows. s denotes the sender node; r denotes the receiver node; Ks,pub and Ks,pri denote the public and private keys of node s, respectively; E(m,K) denotes the public key encryption algorithm with a key K on message m, where

m = M +{ID_f } + SN, and M is the original message; ID_f denotes the ID off, which is the node that forwards the message m; SN is the sequence number of the message; and h(m + k) denotes the keyed hash algorithm with a key k on message m, where + denotes the concatenation of strings. It can be seen that any node that handles the message has to append its ID for non repudiation service. The ID is protected together with the forwarded message. Whenever there is a need for a node to initiate a route discovery process, it creates pair wise shared keys with intermediate nodes, hop by hop, until it reaches the destination. First, it picks random number num. Then, it signs num with its private key by using a public key algorithm like RSA. After that, the route discovery message is protected by a keyed hash MAC algorithm such as MD5. Finally, the hash value and signature can now be attached to the route discovery message and sent out to its neighbors. The complete route request (RREQ) packet sent by the node can be summarized as

m+ h(m + num) + E(num, $K_{s,pri}$). (3)

Those who are s's neighbors and have its public key are able to verify the signature and thus decrypt the key in the message. Suppose that z £ $N_1(s)$ is one of s's hop-l neighbors. Whenever there is a need for s to initiate a route discovery process, it picks a key kl at random, which will serve as the shared secret key between s and z. Then, s encrypts the key kl by using its neighbor's public key Kz,pub. After that, it encrypts the above encrypted key by using its own private key Ks,pri. The result serves as a signature for the route discovery message, which is protected by a keyed hash MAC algorithm such as MD5. The complete procedure is called Keyed MD5. The complete RREQ sent by s can be summarized as

m_q + h(m_q + k_1) + E(E($k_{1,}$ Kz,pub), Ks,pri) , for z £ N_1(s) (4)

where mq stands for the message used in RREQ. This way, only the node that has z's private key can read the key k1, the receiving node is also assured that the key and message come from s, and finally, the integrity of message m can be verified by the receiving node after it decrypts the key. Then, z sends back s a route reply (RREP) packet in a similar format

m_p + h(m_p + k_1) + E(E($k_{1,}$ Ks,pub), Kz,pri) , for z £ N_1(s) (5)

where mp stands for the message used in RREP. By decrypting the message and comparing the key, s can authenticate z and distribute a shared key to z. Similarly, s establishes a shared key with each of its hop-I neighbors. Suppose that y £ $N_1(z)$.z c an also similarly find out its hop-I neighbors and also establishes a shared key with each of them. For s to send messages to its hop-2 neighbors, i.e., N_2(s), for example, y, s requests z to forward the message to y. In z's handshaking with y, z can pick s's public key instead of a random key and send it to y. This way, s's public key can be delivered to its hop-2 neighbors. Similarly, s can obtain the public keys of its hop-2 neighbors. By checking the acknowledgement message back from y via z, s can find out all of its hop-2 neighbors N_2(s). Therefore, s can send a message to r £ N_2(s), via z £ $N_1(z)$ in the following in format:

m2 + h (m2 + k1), k1 =: shared key between s and y (6)

where

m2 = m + h(m + k2) + E (E(k2 , K r, pub) K s, pri) for r £ N_2(s) (7)

where k2 is the shared key between s and its hop-2 neighbor r. Similarly, by using the double hash and signature operations, the shared key between s and its hop-n neighbors,

i.e., kn, is created by s and distributed to Nn(s) where n = 2,3, ...

In the above key distribution process, the same message m has been sent to the destination multiple times and protected by different secret keys at each time. This is what we call message redundancy. To utilize the message redundancy, the implementation is simple: each node is required to receive multiple copies of the same route discovery message before sending back an acknowledgment. It is noted that receiving multiple copies, instead of the first copy, incurs overhead to the route discovery process. The number of copies is determined by two factors. The first one is security, i.e., the trustworthiness of the nodes in the network. To build a route with a certain amount of trustworthiness, the destination needs to evaluate more copies in a less-trusted environment than in a more-trusted one. The second one is performance, i.e, the timeout value of the route request message.

VI. TRUST MODELING AND OPTIMAL ROUTING

We define the trustworthiness on a node n by another node x as the probability that n will perform a particular action expected by x, which is denoted as Tx(n), irrespective of the ability to monitor or control n. The trustworthiness can be evaluated by x in terms of its knowledge accumulated during a specific operation period by using weighting average over the trust on each category of actions, including route request, route reply, route error, and data transmission. We assume that during an observation period, x has received a total of m_t message transmissions from n, among which m_c's are found to be correct; the total number of attempted transmissions is ma; and the total number of successful transmissions is ms. Then

$$T_x(n) = \frac{m_c + \epsilon m_s}{m_t + \epsilon m_a} \qquad \textbf{(8)}$$

h 0 l h f h

where 0 < £ < 1 a weighing factor that represents a ratio of the successful transmissions, which reflects the probability that the link correctly works. Here, we adopt a statistical model similar to the one used for measuring link quality in, which is different from the trust level evaluation. The model in (9) not only evaluates the trustworthiness but also partially reflects the link quality. Other more complicated measurements used to model the link quality, such as the collision detection and signal separation technique, and link adaptation and power control algorithm may also be applied to obtain a more accurate trustworthiness value. Denote by T_x(n;j) the trustworthiness in node n, which is assigned by node x during the jth trustworthiness updating cycle. Every time a new observation comes in, the node updates its repository and calculates a trustworthiness value by using a weighted average or moving average model. Assume that during the jth trustworthiness updating cycle the measurement of T_x(n;j) is denoted as ~T_x(n;j) which is computed based on n's current behavior when x checks the correctness and validity of the messages that come from n. During the (j + l)th trustworthiness updating cycle, these values are used to obtain an estimate of the trust-worthiness, which is denoted as ^T_x(n;j) To obtain a smooth estimation, we use a moving average model

T_x (n;j+1)=α T_x (n;j) +(1- α) T_x (n;j) for n £ N_1(x) (9)

where 0 < α < I is a weighting factor used to tradeoff between current measurement value and previous estimate. Consider a path p £ Ps→x, where Ps→x is the set of paths that start from a source node s to a destination node x, i.e., Ps→x = {all paths from s to x}. Denote by Tx(P;j) the trustworthiness of the path assigned by node x. Thus, the path trustworthiness can be expressed as

Tx(p;j)= Π_{nEp} Tx(n;j) (10)

If Y is on the route from s to x, i.e., y £ p, then y £ N)(x). Denote by pI the path from s to y. Therefore, x can build up its trustworthiness on a path based on its trustworthiness on its neighboring nodes. The relationship in (10) is also used as a routing metric for a node to make routing decisions

VII. CONCLUSION

This paper has proposed an attack detection and defense mechanism by using both the route redundancy in ad hoc networks and the message redundancy in topology discovery of the routing protocols. This paper also develops an optimal routing algorithm by combining both trustworthiness and performance. To our knowledge, this is the first secure routing that quantitatively considers not only the detection of difficult internal attacks but the network performance as well. The proposed attack detection and routing algorithms can be integrated into existing routing protocols for MANETs, such as AODV and DSR. In this and other secure routing protocols, the computational burden at each node is still a major issue in deployment. It requires both analytical investigations and engineering considerations. For example, how many neighbors should a node have without degrading network performance and security? How many copies should a node receive before sending back an acknowledgement? Current paper considers the link performance as a routing metric. Considering the mobility is expected to increase the prediction accuracy and thus reduce the link breakage rate during deployment. All these problems will further be investigated in future work

REFERENCES

[1] C.E. Perkins, Ad Hoc Networking. Addison Wesley Professional, Dec. 2000.

[2] D.B. Johnson, “Routing in Ad Hoc Networks of Mobile Hosts,” Proc. IEEE Workshop Mobile Computing Systems and Applications, Dec. 1994.

[3] L. Buttyan and J. Hubaux, "Nuglets: A Virtual Currency to Stimulate Cooperation in Self-organized Ad hoc Networks". Swiss Federal Institute of Technology, Lausanne DSC/2001/001, 2001.

[4] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A Simple, Cheat-proof, Credit-based System for Mobile Ad-hoc Networks". INFOCOM 2003, pp. 1987 - 1997, 2003.

[5] Y. Liu and Y. R. Yang, "Reputation Propagation and Agreement in Mobile Ad-hoc Networks". Proceedings of IEEE

Wireless Communications and Networking (WCNC 2003), New Orleans, USA, pp. 1510-1515, 2003

[6] S. Yan Lindsay, Y. Wei, H. Zhu and K. J. R. Liu, "Information Theoretic Framework of Trust Modeling and Evaluation for Ad Hoc Networks". IEEE Journal on Selected Areas in Communications, 24(2), pp. 305-317, 2006

[7] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne:: A Secure Ondemand Routing Protocol for Ad Hoc Networks". Proceedings of International Conference on Mobile Computing and Networking, pp. 12-23, Atlanta, USA, 2002.

[8] P. Papadimitratos and Z. J. Haas, "Secure Routing for Mobile Ad hoc Networks". Proceedings of SCS Communication Networks and Distributed Systems Modeling and Simulation Conference, USA, 2002

[9] L. Zhou and Z.J.Haas, "Securing Ad Hoc Networks". IEEE Network,13(6), pp. 24-30, 1999

[10] H. Luo, P. Zerfos, J. Kong, S. Lu and L. Zhang, "Self-securing Ad Hoc Wireless Networks". IEEE ISCC, 2002

[11] Aldar C-E Chan, “Distributed Symmetric Key Management for Mobile Ad hoc Networks”. INFOCOM 2004, China, pp. 2414-2424, 2004

[12] Laura Marie Feeney. A taxonomy for routing protocols in mobile ad hoc networks. Technical report, Swedish Institute of Computer Science, Sweden, 1999.

[13] Jochen Schiller. Mobile Communications. Addison-Wesley, 2000.

[14] Xiaoyan Hong, Kaixin Xu, and Mario Gerla. Scalable routing protocols for mobile ad hoc networks. 2002.

[15] Elizabeth M. Royer and Chai-Keong Toh. A review of current routing protocols for ad hoc mobile wireless networks. Technical report, University of California and Georgia Institute of Technology, USA, 1999.

[16] Charles E. Perkins. Ad Hoc Networking. Addision Wesley, 2001.

[17] Tseng Y.C., Shen C.C, and Chen W.T. Mobile ip and ad hoc networks: An integration and implementation experience. Technical report, Dept. of Comput. Sci. and Inf. Eng., Nat. Chiao Tung Univ., Hsinchu,, Taiwan, 2003.

[18] Danny D. Patel. Energy in ad-hoc networking for the pico radio. Technical report.

[19] Guoyou He. Destination-sequenced distance vector (DSDV) protocol. Technical report, Helsinki University of Technology, Finland

[20] Charles E. Perkins and Elizabeth M.Royer. Ad-hoc on-demand distance vector routing. Technical report, Sun Micro Systems Laboratories, Advacnced Development Group, USA.