A New Security Protocol for Mobile Ad Hoc Networks

K. Sahadevaiah
Department of CSE,
JNTUK College of Engineering,
Kakinada - 533003, AP

Prasad Reddy P.V.G.D.
Department of CSSE,
AU College of Engineering,
Visakhapatnam - 5300003, AP

G. Narsimha
Department of CSE,
JNTUH College of Engineering,
Jagityala - 505501, AP

ABSTRACT
A mobile ad hoc network (MANET) is a self-organized wireless short-lived network consisting of mobile nodes. The mobile nodes communicate with one another by wireless radio links. The unconstrained nature of a wireless medium of MANETs allows the attackers for interception, injection, and interference of communication among nodes. Various secure routing protocols, such as SAR, ARAN, SAODV, SRP, ARIADNE, SEAD, SMT, SLSP, CONFIDANT, etc. are existing in the literature. But these protocols are either too expensive or have unrealistic requirements. They consume a lot of resources. Security extensions for existing routing protocols do not contain important performance optimizations. In this paper, we propose a new security protocol, called cryptographic hybrid key management for secure routing in MANETs. The proposed security protocol has been implemented in Java SE 6 with light weight Bouncy Castle 1.6 API and empirically evaluated its performance via a security analysis and simulation assessments. The results obtained by the proposed approach have been compared with the results of other approaches. Simulation assessments have shown that the proposed approach has outperformed the others, and is a more effective and efficient way of providing security in MANETs.

Keywords
mobile ad hoc networks, self-organization, cryptography, network security, key management, key authentication, key repository, certificate repository, trust graphs.

1. INTRODUCTION
Mobile wireless networking is an emerging technology to access information and services electronically at anytime regardless of their geographic positions. A Mobile Ad hoc NETwork (MANET) is a self-organized wireless short-lived network consisting of mobile nodes. The mobile nodes communicate with one another by wireless radio links without the use of any pre-established fixed communication network infrastructure. Self-organizing means that MANETs have the ability to spontaneously form a network of mobile nodes or hosts, combined together or divided into separate networks on-the-fly, and handle the joining or leaving of the nodes in the network on its own. The major objectives of self organized MANET are: scalability, reliability, and availability. Each mobile node acts as both a host and a specialized router to transfer information to other mobile nodes. The success of the nodes' communication in radio range highly depends on the dynamic discovery of other nodes' cooperation. Typical MANET nodes are Laptops, PDAs, Pocket PCs, Cellular Phones, Internet Mobile Phones, and Palmtops. These devices are typically lightweight and battery operated. The main characteristics of MANETs are: lack of centralized control, lack of association among nodes, rapid mobility of hosts, dynamically varying network topology, shared broadcast radio channel, insecure operating environment, physical vulnerability and limited availability of resources, such as processor capacity, storage capacity, battery power, and bandwidth [2, 6, 7, 8]. The domain of applications for MANETs is distinctive, ranging from large-scale, mobile, highly dynamic networks to small, static networks which are constrained by power sources. Significant examples include establishing survivable, efficient, dynamic communication for: network-centric military/battlefield environments, emergency/rescue operations, disaster relief operations, intelligent transportation systems, conferences, fault-tolerant mobile sensor grids, smart homes, patient monitoring, environment control, and other security sensitive applications. Most of these applications demand a specific security guarantees and reliable communication [2, 5, 7, 9].

Node mobility in a MANET poses many security problems. The mobile nodes are vulnerable to different types of security attacks than conventional wired and wireless networks. This is due to their open medium, dynamic network topology, absence of central administration, distributed cooperation, constrained capability, and lack of clear line of defense. An extensive number of research works on designing the various routing protocols (proactive, reactive, and hybrid) has been proposed in the literature and widely evaluated for efficient routing of packets. However, these routing protocols do not address possible threats aiming at the disruption of the protocol itself and often are vulnerable to node misbehavior. With the lack of a priori trust between nodes, current ad hoc routing protocols are completely insecure and optimized to disseminate routing information more efficiently as the network topology changes. MANETs need secure routing protocols to prevent possible security attacks. Various secure routing protocols, such as SAR, ARAN, SAODV, SRP, ARIADNE, SEAD, SMT, SLSP, CONFIDANT, etc. are existing in the literature. But these protocols are either too expensive or have unrealistic requirements. They consume a lot of resources, and delay or even prevent successful exchanges of routing information. Security extensions for existing routing protocols do not contain important performance optimizations. Inclusion of optimistic approaches provides a better trade-off between security and performance. Resource limitations of mobile devices, such as memory, computation, communication and energy, need to be carefully considered in the solution. The major aim of this paper is to examine the deficiencies of the existing secure routing protocols and propose a new security protocol called - cryptographic hybrid (symmetric/ asymmetric) key management for secure routing in MANETs for handling a large number of mobile nodes. The paper involves the design of the proposed security protocol and investigates, in detail, the performance of the proposed security protocol against various known and unknown malicious node attacks. The proposed security protocol solutions rely on private-public key cryptography and digital signatures to achieve the security goals like message integrity, data confidentiality, and end-to-end authentication. In the proposed scheme, the nodes need not be responsible for issuing other nodes' certificates. Every

intermediate node checks the neighbor's digital signatures, which guarantee that no single node modifies the public key certificate information during the distribution process. The reason is that the certificates are distributed securely to the neighboring nodes with the symmetric key encryption. Furthermore, the method does not involve any trusted authority, not even in the system initialization phase. The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 provides the description of the proposed security protocol. The simulation results and the security analysis are described in Section 4. Finally, Section 5 concludes the paper.

2. RELATED WORK

Peer-to-peer or pairwise key management protocols are designed for both authority-based MANETs and fully self-organized MANETs [15, 16]. Authority-based MANETs support applications that demand the use of an offline authority. The nodes do have pre established relationships. The trusted authority set up the nodes with shared cryptographic keying material prior to network formation and form strong security associations between nodes. After network formation, each node becomes its own authority domain and distributes its certificate to nodes within its transmission range. Fully self-organized MANETs do not have any form of online or offline authority. These networks are created solely by the end-users in an ad hoc fashion. The users forming the ad hoc networks have no pre established relationships and, therefore, share no common keying material on their nodes. Users have to set up security associations between them, after network formation, without the aid of a common online/offline trusted third party (TTP). S. Capkun, L. Buttyan, and J. P. Hubaux (2002, 2003) [15, 18] proposed a self-classified public-key management system for MANETs. The protocol is an extension of PGP and permits nodes to create, store, distribute, and withdraw their public-private keys & public key certificates without the help of any trusted authority or fixed server in a fully self-organized manner. S. Capkun, J. Hubaux, and L. Buttyan (2006) [16] proposed a straightforward technique, called mobility based key management, to provide how mobility helps to start up security associations for protected routing in peer-to-peer mobile networks employed either with symmetric or with asymmetric cryptography.D. Choi, Y. Lee, Y. Park, S. Jin, and H. Yoon (2008) [17] proposed a scheme, called efficient and secure self-organized public key management for MANETs that comprises of (1) handshaking (HS) and (2) certificate request/reply (CRR) procedures. H. Dahshan and J. Irvine (2009) [31, 32] proposed a trust model based on the existence of public-key certificates as bindings of the public keys creating a small number of trust relations between neighboring nodes through the network initialization phase.

When the scale of the network becomes larger, the following are the limitations/drawbacks of the previous works: An Efficient Authenticity Problem - The system is represented as a directed certificate graph, in which vertices denote users and edges denote certificates. Public key authentication is performed via certificate chaining. To authenticate public key via certificate chain, more than one certificate needs to be verified. However, this scheme suffers from the delay and the large amount of traffic required collecting certificates. Certificate conflicting is an example of a potential problem. The certificate graph may not be strongly connected, since nodes within one component may not be able to communicate with ones in different components is another example of a potential problem. The Security Problem - Two nodes merge their local certificate repository and attempt to find a chain of certificates connecting them. As the length of the certificate chain increases, the trustworthiness of the public key obtained through the chain might be decreased. Hence, the system might become vulnerable to attacks. This method is self-organized, but its transitivity of trust property is vulnerable to an active attack. The Overhead Problem - Each node in the network maintains two kinds of repositories, non-updated certificate repository and updated certificate repository. Multiple certificates are issued. The approximate global certificate graph is stored in the non-updated certificate repository and the certificates required to be updated periodically are stored in the updated repository. The main problems with the previous schemes are large communication overhead for the certificate repository of a mobile node to store an approximate global certificate graph. The Side Channel/ Radio Channel Problem - The previous schemes consist of: handshaking (HS) and certificate request/reply (CRR) procedures. In HS, a node attains the public key of the forthcoming node through a safe side channel, such as an infrared interface. In CRR, a node request certificates of a remote node through a radio channel to the nodes that it has handshaked. The drawback is the use of the side channels, radio channels, and the threshold cryptography.

3. PROPOSED SECURITY PROTOCOL

The proposed security protocol, called the *cryptographic hybrid key management for secure routing in MANETs*, provides the self-organized behavior by sharing the public keys and self-signed certificates among the nodes to form the network with an initial trust phase. The network nodes of the proposed scheme need not be responsible for issuing other nodes' certificates. Every intermediate node checks the neighbor's digital signatures, which guarantee that no single node may modify the public key certificate information during the distribution process. The reason is that the certificates are distributed securely to the neighboring nodes with the symmetric key encryption. The main goal of the proposed scheme is to provide a secure environment for transmission of messages from source to destination, where the source allows encrypting the messages that will be decrypted at destination only. The network operation of all nodes of the proposed scheme starts in a secure environment, called as an initial trust phase. Each node becomes a neighbor to any other node which covers its radio range and offers its public key. This process takes place only to share the public keys among the neighboring nodes. After completion of the trust phase, the system quits the secure environment because every node contains the public keys of all participated nodes in the network. Then, the nodes get dispersed in a random order. Each node has to keep the trust phase information for future authentication. After successful exchange of the public keys, the certificates encrypted with destination public key are issued with a limited validity period that contains its issuing and expiration times. When a certificate expires, its issuer issues an updated version of the same certificate which contains an extended expiration time. After successful exchange of public keys and certificates of neighbors, each node constructs its own key repository and certificate repository. The certificate repository contains only the neighbor certificates. The shared key repository contains all the public keys of the nodes in the network. Key repositories will be shared in an encrypted form along with the certificates. Each node, also, builds its own shared key repository and trust graph. The public keys and the certificates of the system are represented as a directed graph, called the *trust graph*. The vertices of the trust graph represent public

keys and the edges represent certificates. A trust graph is useful for finding the efficient route. This trust graph will be saved as master graph (MG). Whenever a change occurs in the shared key repository, immediately it informs the other neighbor nodes about the updated shared key and the trust graph. Based on the trust graph and the public key expire time of each node in the existing path, the scheme finds the efficient path for sending the message. To secure a MANET, a security protocol must satisfy the attributes: confidentiality (privacy), availability, integrity, authenticity and non-repudiation. The proposed scheme achieves the confidentiality by encrypting the message with the sender's AES symmetric key generated for that message, thereby making it impossible for the attacker to get useful information from the data overheard. The receiver's RSA public key is used to encrypt the AES secret key. Then, the message digest is encrypted with the sender's RSA private key so that all the security goals are achieved.The algorithmic operations of the proposed security protocol are as follows:

Step1: //Each device creates public-private key pairs and public key certificates
Generate $K_S(X), K_P(X), C_X \ \forall X \in N$;
where N is number of devices, $K_S(X)$ is secret key of device X, $K_P(X)$ is public key of device X, and C_X is certificate of device X.
Initialize
$NKR_X, NCR_X, SKR_X, SMR_X, RMR_X$;
where NKR_X is neighbors key repository of device X, NCR_X is neighbors certificate repository of device X, SKR_X is shared key repository of device X, SMR_X is sent messages repository of device X , and RMR_X is receiving messages repository of device X.

Step2: //Exchange of public keys and construction of public key repository
Issue $K_P(X) \Rightarrow Z$; where $\forall Z \in NBR(X)$ and NBR(X) is neighbors of node X
Store $NKR_X \Leftarrow K_P(Z)$; // Receives $K_P(Z)$ and stores it in NKR_X
Step3: //Exchange of public key certificates and construction of certificate repository

$$PKT_X = (Enc_{AES}(C_X), Enc_{K_S(X)}(Digest(C_X)), Enc_{K_P(Z)}(AES));$$

where PKT_X is a data packet of device X, Enc_{Key} is encryption with the key.

Issue $PKT_X = Z$;

$$PKT_Z = (Dec_{K_S(X)}(AES), Dec_{AES}(C_Z), Dec_{K_P(Z)}(Digest(C_Z)));$$

where Dec_{Key} is decryption with the key. Receives PKT_Z by device X and decrypt it and then verify the authentication.
if (decrypted value of $Digest(C_Z) \equiv$ value of $Digest(C_Z)$ calculated at X) then $NCR_X \Leftarrow C_Z$
Step 4: //Each device constructs and exchanges of shared key repository and trust graph
$SKR_X \Leftarrow NKR_X$; Send $SKR_X \Rightarrow Z$;
Receive SKR_Z andupdate SKR_X until SKR_X contains **N** public keys.
Construct TG_X and Send $TG_X \Rightarrow Z$; TGX is trust graph of node X.
Receive TG_X and update TG_X;
Step 5: //To find shortest route using fisheye state routing protocol
if
$((SKR_S(D)\,RemainingKeyExpirationTime) > 60\,seconds)$
$RemainingKeyExpirationTime[X] = SKR_S(X)KeyGenerationTime - CurrentTime;$
where $\forall S, D, X \in N$
// Initialization of Dijkstra's variables
Initialize $State[X]$ with infinite length, label status as true; where $State[X]$ is the state of device either *visited or not* while detecting the path.
$KeyExpirationTime[X] = KeyRemainingExpirationTime[X];$
if $(X \equiv Y) adj[X][Y] = 0$;
else { $adj[X][Y] = 1$; Count the total number of edges; where $\forall X, Y \in N$ }
$State[D].length = 0, State[D].status = false, Current = D$
; where $\forall Current \in N$
repeat {
if $(adj[Current][X] \neq 0 and State[X])$ is not visited; where $\forall Current, X \in N$
{if ((State[Current].length + adj[Current][X] ≡ State[X].length) && KeyExpirationTime [Current] + KeyRemainingExpirationTime[X] >= KeyExpirationTime[X]) Assign label to State[X];
if (State[Current].length + adj[Current][X] < State[X].length) Assign label to State[X];
}
if $\exists$ X with minimum length, then Current = X;
} until source node is detected/found;
repeat { $path_S(D)$ = State[X].next; } until destination is reached;
if determined path contains D then return $Path_S(D)$, where $\forall D \in Path_S(D)$;

Step 6: //The plain text transmission from a given source to destination
$SMR_S \Leftarrow (M_{S,D}, C_Z \oplus Path_S(D))$; where $\forall S, D, Z \in N$, $Path_S(D)$ is the shortest path from source(S) to destination (D), $M_{S,D}$ is a message sent from source(S) to destination(D), and SMR_S is the sent message repository at source.
$PKT_S \Leftarrow (M_{S,D}, C_Z \oplus Path_S(D))$; where PKT_S is a data packet at the source and $\forall S, D, Z \in Path_S(D)$
Send $PKT_S \Rightarrow Z$ which in turn forwards $PKT_Z \Rightarrow Y$ until it reaches the destination, where $\forall S, Z, Y \in Path_S(D)$
$RMR_D \Leftarrow (M_{S,D}, C_D \oplus Path_S(D))$;
where $\forall S, D \in Path_S(D)$ and RMR_D is received message repository at destination

Step 7: //Behavior of security attacks
if (*device X is a wormhole attacker*), creates tunnel path and send $PKT_X \Rightarrow Y$; which in turn forwards $PKT_Y \Rightarrow W$ until it

reaches to another attacker, where tunnel path $_{W1}$(W2) is a tunnel path created between wormhole1 to wormhole2, $\forall X \in Path_S(D),\ \forall Y, W \in tunnelpath_X(Z)$;

if (*device X is a man-in-the-middle attacker*), PKT_X = ($M_{S,D}$ +"Invalid random number") corrupts data and forwards $PKT_X \Rightarrow Y$,

where $\forall X \in N,\ \forall X \in NBR(Y), \forall Y \in Path_S(D)$;

if (*device X is a denial of service attacker*) No forwarding of data, where $\forall X \in Path_S(D)$;

if (*device X is a sybil attacker*) Send invalid P_X with spoofed ID's of Z where $\forall X \in N, \forall Z \in NBR(X)$;

if (*external misbehavior attacker enters into the network and try for connection establishment within the devices radio range*) then Raises Authentication error "Authentication Failed";

Step 8: //Secure plain text transmission from source to destination

$SMR_S \Leftarrow (M_{S,D}, C_Z \oplus Path_S(D))$; where $\forall S, D, Z \in N$, $Path_S(D)$ is the shortest path from source(S) to destination (D), $M_{S,D}$ is a message sent from source(S) to destination(D), and SMR_S is the sent message repository at source.

$$PKT_S = (Enc_{AES1}(M_{S,D}), Enc_{K_S(S)}(Digest(M_{S,D})), Enc_{K_P(D)}(AES1), Enc_{AES2}(C_X \oplus Path_S(D)), Enc_{K_S(S)}(Digest(C_X)), Enc_{K_P(X)}(AES2))$$

$\forall S, X \in Path_S(D), \forall X \in NBR(S)$

Send $PKT_S \Rightarrow X$ which in turn sends $PKT_X \Rightarrow Z$, where $\forall S, X, Z \in Path_S(D), \forall X \in NBR(S)$ // Certificate is decrypted at each hop and PKT_Z is forwarded until it reaches the destination.

$$PKT_D \Leftarrow (Dec_{K_S(D)}(AES2), Dec_{AES2}(C_D \oplus Path_S(D)), Dec_{K_P(Z)}(Digest(C_D)))$$

; where PKT_D is a packet decryption at destination and

$\forall D \in Path_S(D), \forall Z \in NBR(D)$

if (decrypted value of $digest(C_D) \equiv$ values of $digest(C_D)$ calculated at D) then

$$PKT_D \Leftarrow (Dec_{K_S(D)}(AES1), Dec_{AES1}(M_{S,D}), Dec_{K_P(S)}(Digest(M_{S,D})))$$

// Neighbor authentication

if (decrypted value of $digest(M_{S,D}) \equiv$ value of $digest(M_{S,D})$ calculated at D) then

$RMR_D \Leftarrow (M_{S,D}, C_D \oplus Path_S(D))$; //Source authentication where $\forall S, D \in Path_S(D)$ and RMR_D is received message repository at destination.

4. EXPERIMENTAL RESULTS AND ANALYSIS

This section describes the experimental network scenarios and the security analysis of simulation. The proposed scheme has been implemented in Java SE 6 with lightweight Bouncy Castle 1.6 API. The security protocol solutions, proposed in the present work, rely on reliable security mechanisms - private and public key cryptography (Advanced Encryption Standard (AES), RSA, X.509 certificates, digital signatures) and secure hash based message authentication codes (SHA1). The use of cryptographic principles takes more time to encrypt and decrypt at every node. To avoid this, we have used the hybrid encryption techniques both the symmetric and asymmetric algorithms. Simulation results have shown that the proposed scheme resists against malicious nodes, which sign and issue false public key certificates for other nodes in the network, with low implementation complexity. The results obtained by the proposed approach are compared with the results of other approaches. When the malicious nodes or radio range/ power range is increased in the network, the certificate successful rate is better compared with previous approaches. It was found that the proposed approach had outperformed the others. We analyze the security of a proposed scheme rigorously via the impact of malicious nodes on successful certificate rate, impact of network partitioning (radio range) on successful certificate rate, and the impact of different types of security attacks on secure routing. To perform the security analysis, the following assumptions are made about capabilities of the attacker:

1. The attacker listens and makes a record of all the traffic in the network.

2. When the node is captured, all the information stored in a node is known by the attacker.

3. The attacker captures a set of nodes selectively or randomly in a network.

4.1 Comparison with Previous Schemes

In this section, we compare the performance of the proposed scheme with those for the previous schemes, namely, S. Capkun, L. Buttyan, and J. P. Hubaux [15] and D. Choi, Y. Lee, Y. Park, S. Jin, and H. Yoon [17]. The comparisons are made on successful certificate rate against the impact of malicious nodes as well as radio range (power range).

A network is deployed with 100 nodes. The certificate successful rate is measured by increasing the number of malicious nodes. As shown in Table 4.1, the proposed approach of certificate successful rate is compared with previous approaches S. Capkun et al.[15] and D. Choi et al.[17].

Table 4.1: Comparison of impact of malicious nodes

Malicious Nodes	S. Capkun et al. [15]	D. Choi et al. [17]	Proposed Approach
10	79.5133	83.4125	86.0359
20	74.2185	78.2455	81.4739
30	62.3956	65.2459	71.1487
40	53.272	54.649	58.3952
50	45.532	46.8246	49.6549
60	36.2857	38.8594	42.4269
70	27.948	29.2346	35.9162
80	18.8724	20.1368	25.2325

As shown in Figure 4.1, the resulting data are plotted using MATLAB 7.6 [33, 34]. Each data point in the resulting graph is an average of four program runs with an identical configuration of 100 nodes, but different randomly generated mobility patterns.

When the malicious nodes are increased to 80, the certificate successful rate of the proposed approach is ***25.2325%*** compared to S. Capkun et al. (18.8724%) and D. Choi et al. (20.1368%).

A network is deployed with 100 nodes. The certificate successful rate is measured by increasing the power range or radio range. As shown in Table 4.2, the proposed approach of certificate successful rate is compared with previous approaches S. Capkun et al.[15] and D. Choi et al.[17].

As shown in the Figure 4.2, the resulting data were plotted using MATLAB 7.6 [33, 34]. Each data point in the resulting graph is an average of four program runs with an identical configuration of 100 nodes, but different randomly generated mobility patterns. When the radio ranges are increased to 240, the certificate successful rate of the proposed approach is ***86.187***% compared to S. Capkun et al. (69.0164%) and D. Choi et al. (75.4553%)

4.2 Impact of Security Attacks on Routing

then attempts to receive all the packets destined for the legitimate node. In a Denial of Service (DoS) attack, an adversary always attempts to prevent legitimate and authorized users of network services from accessing those services, where legitimate traffic cannot reach the target nodes. DoS attacks are

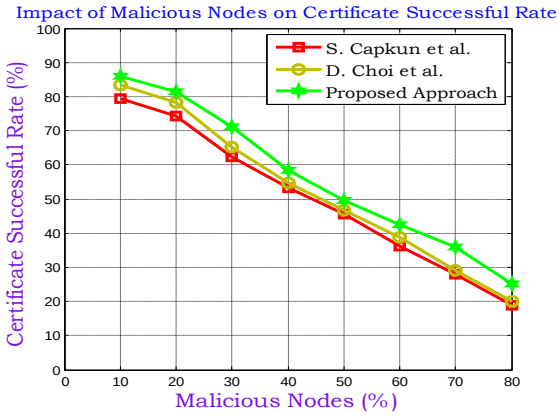

Figure 4.1: Impact of malicious nodes on certificate successful rate

Table 4.2: Comparison of impact of radio ranges

Radio Range	S. Capkun et al. [15]	D. Choi et al.[17]	Proposed Approach
100	0.3651	1.2198	1.9079
120	1.162	2.1225	2.7515
140	2.485	6.8831	7.3473
160	17.5871	19.5616	23.5555
180	42.5684	44.4114	48.496
200	62.3956	68.2459	70.1487
220	67.1684	69.9248	75.1363
240	69.0164	75.4553	86.187

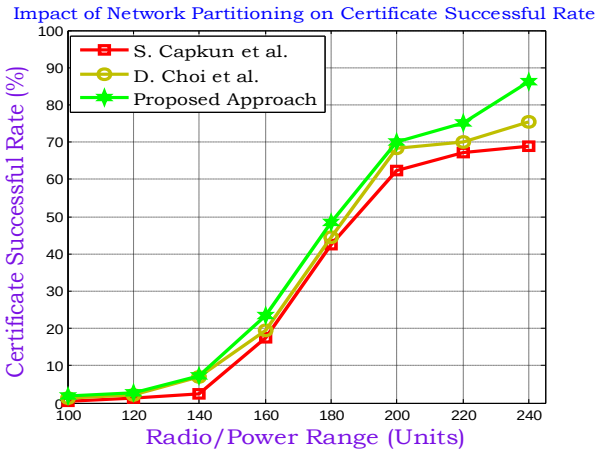

Figure 4.2: Impact of radio range on certificate successful rate

against CPU power, battery power and transmission bandwidth. In an Information Disclosure attack, a compromised node attempts to reveal confidential or important information regarding the network topology, geographic locations of nodes, or optimal routes to unauthorized nodes in the network. In a Wormhole attack, a malicious node captures packets from one location in network and tunnels these packets to other malicious node at other location. The comparisons are made on routing overhead against the security attacks. Table 4.3 shows the data routing time, in seconds, for different network sizes. Each data point in the resulting table is an average of four program runs with an identical configuration of various network sizes, but different randomly generated mobility patterns. When the network size is increased to 100, the various routing times (in seconds) are: Plaintext Routing Time (PRT): 5.5236, Secure text Routing Time (SRT): 7.6988, Plaintext Routing Time with Wormhole Attack (PRTWH): 5.8674, Plaintext Routing Time with Man-In-The-Middle Attack (PRTMIM): 5.9766, and Plaintext Routing Time with DoS Attack (PRTDoS): 10.4848.

Table 4.3: Comparison of routing overhead against security attacks

Network Size	PRT	SRT	PRT WH	PRT MIM	PRT DoS	PRT SY
10	2.3361	2.7023	2.3666	2.3283	2.8986	4.2782
20	2.8882	3.2604	2.9144	2.7124	4.7106	5.0821
30	3.0654	3.7261	3.3672	3.2012	5.6662	5.8576
40	3.3694	4.0272	3.5562	3.5634	6.3176	6.1542
50	3.5422	4.3454	3.7696	3.8026	6.6574	6.4095
60	3.9118	4.6172	3.9742	4.0972	7.0963	6.9196
70	4.1523	5.2788	4.4108	4.5938	7.8794	7.2726
80	4.7541	5.8622	5.0438	5.0034	8.5642	7.6352
90	5.1064	6.7014	5.4328	5.4476	9.2448	7.8924
100	5.5236	7.6988	5.8674	5.9766	10.4848	8.2566

The resulting data of the Table 4.3 are plotted using MATLAB 7.6 [33, 34] and is shown in Figure 4.3.

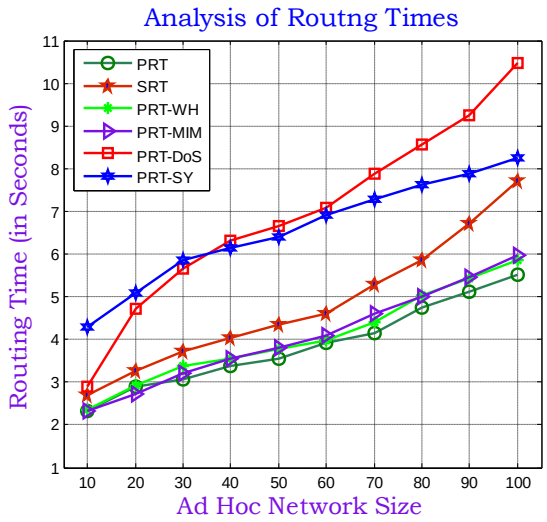

Figure 4.3: Analysis of routing overhead with and without security attacks

5. CONCLUSION

The proposed security protocol has been implemented in Java SE 6 with lightweight Bouncy Castle 1.6 API and empirically evaluated its performance via a security analysis and simulation assessments. Simulation results have shown that the proposed scheme resists against malicious nodes, which sign and issue false public key certificates for other nodes in the network, with low implementation complexity. The results obtained by the proposed approach have been compared with the results of other approaches. Certificate successful rate is better when compared with previous approaches by increasing the number of malicious nodes or by increasing radio range/ power range. It has been found that the proposed approach has outperformed the others, and is a more effective and efficient way of providing security in MANETs.

6. REFERENCES

[1] C.K. Tok, Ad Hoc Mobile Wireless Networks: Protocols and Systems, Pearson Education, pp. 28-30, 2002.

[2] X. Cheng, X. Huang and D.Z Du, Ad Hoc Wireless Networking, Kluwer Academic Publishers, pp. 319-364, 2006.

[3] C. Siva Ram Murthy and B.S Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Pearson Education, 2006.

[4] F. Anjum and P. Mouchtaris, Security for Wireless Ad hoc Networks, John Wiley & Sons, 2007.

[5] Prasant Mohapatra and Srikanth V. Krishnamurthy, Ad Hoc Networks: Technologies and Protocols, Springer International Edition, 2005.

[6] C E. Perkins: Ad Hoc Networks, Addition Wesley, 2001.

[7] S. Basagni, M. Conti, S. Giordono and I. Stojmenovic: Mobile Ad Hoc Networks, IEEE Press Wiley, New York, 2003.

[8] A. J. Menezes, P. C. V. Oorschot, and S. A. Vanstone: Handbook of Applied Cryptography, CRC Press, 1996.

[9] L. Zhou and Z. J. Haas: Securing Ad Hoc Networks, IEEE Network Magazine, Vol. 13, No.6, pp. 24–30, 1999.

[10] H. Luo, J. Kong, P. Zerfox, S. Lu, and L. Zhang, URSA: Ubiquitous and Robust Access Control for Mobile Ad Hoc Networks, IEEE/ACM Transactions on Networking, Vol.12, No.6, pp.1049–1063, 2004.

[11] B. Lehane, L. Doyle, and D. O'Mahony, Shared RSA Key Generation in a Mobile Ad Hoc Network, Proceedings of IEEE Military Communications Conference (MILCOM), Vol.2, pp.814–819, 2003.

[12] B. Zhu, F. Bao, R.H. Deng, M.S. Kankanhalli, and G. Wang, Efficient and Robust Key Management for Large Mobile Ad Hoc Networks, Computer Networks - Elsevier, Vol.48, pp.657–682, 2005.

[13] M. Narasimha, G. Tsudik, and J. Yi, On the Utility of Distributed Cryptography and P2P and MANETs: The Case of Membership Control, Proceedings of IEEE International Conference on Network Protocols (ICNP), pp.336–345, 2003.

[14] S. Jarecki, N. Saxena, and J.H. Yi, An Attack on the Proactive RSA Signature Scheme in the URSA Ad Hoc Network Access Control Protocol, Proceedings of ACM Workshop on Security of Ad Hoc and Sensor Networks, pp.1–9, 2004.

[15] S. Capkun, L. Buttyan, and J. P. Hubaux, Self-Organized Public-Key Management for Mobile Ad Hoc Networks, IEEE Transactions on Mobile Computing, Vol.2, No.1, pp.52–64, 2003.

[16] S. Capkun, J. P. Hubaux, and L. Buttyan, Mobile Helps Peer-to-Peer Security, IEEE Transactions on Mobile Computing, Vol.5, No.1, pp.43–51, 2006.Daeseon CHOI, Younho LEE, Yongsu PARK, Seung-hun JIN, and Hyunsoo YOON, Efficient and Secure Self-Organized Public Key Management for Mobile Ad Hoc Networks, IEICE Transactions on Communications,Vol.E91– B, No.11, pp. 3574-3583, 2008.

[17] S. Capkuny, L. Buttyan and J. Hubaux, Small Worlds in Security Systems: an Analysis of the PGP Certificate Graph, New Security Paradigms Workshop 2002, Norfolk, Viginia 2002.

[18] C. Gandhi and M. Dave, A review of security in mobile ad hoc networks, IETE Technical Review, Vol. 23, No. 6, pp 335-344, 2006.

[19] S Marti, T.J. Giuli, K. Lai and M. Baker, Mitigating Routing Misbehavior in Mobile Ad hoc Networks, 6th ACM Annual International Conference on Mobile Computing and Networks (MOBICOM 2000), pp. 255-265, Boston, USA 2000.

[20] 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 (CNDS 2002), pp. 27-31, San Antonio, USA 2002.

[21] H. Deng, W. Li and D.P. Agrawal, Routing Security in Wireless Ad hoc Networks, IEEE Communications Magazine, pp. 70-75, 2002.S. Gupta and M. Singhal,

Secure Routing in Mobile Wireless Ad hoc Networks, Ad Hoc Networks, Vol.1, pp. 151-174 2003.

[22] K. Sanzgiri, B. Dahill, B.N. Levine, C. Shields and E. M. Belding-Royer, A Secure Routing Protocol for Ad hoc Networks, Proceedings of 10th IEEE International Conference on Network Protocols (ICNP2002), pp. 78-87, Paris, France, 2002.

[23] L.M. Kornfelder, Toward a Practical Public-Key Cryptosystem, Bachelor's Thesis, Department of Electrical Engineering., Massachusetts Institute of Technology, Cambridge, 1978.

[24] Allen C. Sun: Design and Implementation of Fisheye Routing Protocol for Mobile Ad Hoc Networks, Massachusetts Institute of Technology, USA, 2000.

[25] R. Sedgewick: Weighted Graphs, Addison-Wesley, chapter 31, 1983.

[26] Internet X.509 Public Key Infrastructure Certificate and CRL Profile - RFC 2459.

[27] Weihong Wang, Ying Zhu, and Baochun Li, Self-Managed Heterogeneous Certification in Mobile Ad Hoc Networks, Proceedings of IEEE Vehicular Technology Conference (VTC 2003), Orlando, Florida, 2003.

[28] Matei Ciobanu Morogan, Sead Muftic, Certificate Management in Ad Hoc Networks, Symposium on Applications and the Internet Workshops (SAINT'03 Workshops), pp. 337, 2003.

[29] H. Dahshan and J. Irvine, On Demand Self Organized Public Key Management for Mobile Ad Hoc Networks, IEEE 69th Vehicular Technology Conference (VTC'09), ISBN: 978-1-4244-2517-4, pp.1-5, 2009.

[30] H. Dahshan and J. Irvine, A Robust Self Organized Public Key Management for Mobile Ad Hoc Networks, Security and Communication Networks, Wiley InterScience, pp. 16- 30, Vol.3, 2009.

[31] T. A. Driscoll, Learning MATLAB, Siam publishers, ISBN: 978-0-898716-83-2, USA, 2009.

[32] Y. Kirani Sigh and B. B. Chaudhuri, MATLAB Programming, Prentice Hall of India, 2007.

[33] E. R. Harold, Java Network Programming, Third Edition, O'Reilly Publishers, ISBN: 0-596-00721-3, 2004.