OA2R: An Optimal Anonymous Authenticated Routing Scheme for MANET's

Ms. Chaithra S, Ms.Chaitra V and Mr. Roopesh Kumar B N

Abstract:- **The popularity of wireless communications recently gave Mobile Ad-Hoc networks (MANET's) a significant researchers attention due to its time and mission critical applications. However its natural advantageous of networking in healthcare and social media environments make them vulnerable to security threats. Many o the existing anonymous routing protocols, however, don't make allowance for authentication thereby making them vulnerable to modification to packet and denial of service attacks. In this paper, we propose the anonymous routing protocol also furnishing authentication in MANET. The main objective is to provide mechanism concealing a real identity of communicating nodes along with providing end-to-end authentication via asymmetric and symmetric key cryptography.**

Index Terms—Anonymous Routing, Authenticated Routing, Mobile Ad hoc Networks.

I. INTRODUCTION

The preponderance of wireless communication recently gave mobile ad hoc networks (MANET) a significant researcher's attention, due to its innate capabilities of instant communication in many time and mission critical applications. However, its natural advantages of networking in healthcare and social media environments make them vulnerable to security threats. Until recently, quite a number of anonymous routing protocols have been proposed. Many of them, however, do not make allowance for authentication. Thus, vulnerabilities such as modifications to packet data and denial of service attacks can be more easily exploited. In this paper, we propose the anonymous routing protocol also furnishing authentication in the mobile ad hoc networks. The main objective is to provide mechanisms concealing a real identity of communicating nodes with an ability of resist to known attacks. The distributed reputation system is incorporated for a trust management and malicious behavior detection in the network. The end-to-end anonymous authentication is conducted in three-pass handshake based on an asymmetric and symmetric key cryptography. After successfully finished authentication phase secure and multiple anonymous data channels are established. The anonymity isguaranteed by randomly chosen pseudonyms owned by a user. In this paper we presented an example of the protocol implementation. Without fixed infrastructure, an ad hoc network is a self constituting wireless network by participating nodes itself. Because the whole nodes have to communicate with each other without access point, in ad hoc network, it is the most important issue to design routing protocol which is path discovery mechanism.In recent years mobile ad hoc networks have received significant researcher's attention due to capabilities of establishing an instant communication in many time-critical and mission-critical applications. Many security protocols have been devised to protect a communication in ad hoc networks [10], [20], however the only few of them address privacy guaranties [3], [4], and [5]. Leaving mobile nodes traceable by wireless traffic and data analysis makes the anonymity support in MANET critical challenge. In this paper we propose a new anonymous authentication protocol for mobile ad hoc networks enhanced with a distributed reputation system. The main objective of this work is to provide protocol with mechanisms concealing real identity communicating nodes and resisting to known attacks [6], [23]. The distributed reputation system is incorporated in order to build and manage trust of communicating nodes. The trust knowledge reflects trustworthy and malicious activity in the network

effectively supporting anonymous authentication and path discovery phases. Proposed protocol delivers secure exchange data links based on on-demand routing approach [1], [7], and [8].

This paper is topology based MANETs, it has chance to get attack in its routing path for this consideration need authenticated based topology routing. Our anonymous communications in MANETs has unidentifiability and unlinkability. Unindentifiability means the source and destination node cannot be identify by the other nodes. Unlinkability means that the route between the source and destination node cannot be linked directly together [3].

The existing protocols are not that much sufficient to anonymous secure delay reduce scheme during packet transmission, such protocols are ANODR, Anon DSR, and Discount-ANONR [4]. After examining these protocols, we find that the objectives for secure routing with reduce delay by Authenticated Anonymous Secure Routing (AASR) with Trust based model. MANETs with Group Signature have both public and private key to select the authenticated mobile nodes in adversarial environment [5] – [6]. The key-encrypted onion to record a discovered route and design an encrypted secret message to verify the RREQ-RREP linkage. The proposing an optimal anonymous authenticated secure routing (OA2R) with calculating Trust value to avoid the delay in anonymous routing path. The following part will present AODV, DSR, TORA and ABR as characteristic protocols of on-demand trend.

The following sections present related work and cover in details a protocol construction supplemented by its anonymous properties analysis. The last section presents some concluding remarks and further research directions.

II. BACKGROUND AND RELATED WORK

A. Anonymous On-Demand Routing Protocols ANODR [12] which is based on onion routing [13, 14] is a anonymous protocol in that each intermediate node encrypt forwarding packet by its public key and decrypt route reply packet by its private key. In route request, packets are added an encrypted layer, that is called boomerang onions. Most of the anonymous routing protocols have similar method with ANODR to find the destination. Cheng et al. proposed ASRP [8] based on public key system. Each intermediate nodes generate their pseudonym to check whether it is intermediate node or not introduced ODAR [15] which is a new concept of anonymous routing protocol. It uses the Bloom Filter for anonymous route maintenance. The Bloom Filter is a data structure that is only available to check whether or not to include an element. The existing anonymous routing protocols [4, 6, 8, 10, 12], not only protocols mentioned above but most of anonymous routing protocols, are not concern with authentication. This means an adversary can illegitimately behave without any restrictions during the routing discovery. In particularly, these protocols are fragile against the DoS (denial of service) attack in that an ad hoc network is the broadcast based wireless network.

In more detail, if the exterior adversary who wants to inflict the overload on the network broadcasts route request packets or re-broadcasts existing control packets, the network resource would be shortly exhausted by maliciously propagated packets. Therefore, for blockading DoS attack, routing protocol must limit that adversary can broadcast packet in its disposition, and also replayed packets must be meaningless to prevent it roaming.

B. Anonymity and Security Primitives

We introduce some common mechanisms that are widely used in anonymous secure routing.

1) Trapdoor: In cryptographic functions, a trapdoor is a common concept that defines a one-way function between two sets [5]. A global trapdoor is an information collection mechanism in which intermediate nodes may add information elements, such as node IDs, into the trapdoor. Only certain nodes, such as the source and destination nodes, can unlock and retrieve the elements using pre-established secret keys. The usage of trapdoor requires an anonymous end-to-end key agreement between the source and the destination.

2) Onion Routing: It is a mechanism to provide private communications over a public network [15]. The source node sets up the core of an onion with a specific route message. During a route request phase, each forwarding node adds an encrypted layer to the route request message. The source and destination

nodes do not necessarily know the ID of a forwarding node. The along the route back to the source.

The intermediate node can verify its role by decrypting and deleting the outer layer of the onion. Eventually, an anonymous route can be established. 3) Group Signature: The group signature scheme [9] can provide authentications without disturbing the anonymity. Every member in a group may have a pair of group public and private keys issued by the group trust authority (i.e., group manager). The member can generate its own signature by its own private key, and such signature can be verified by other members in the group without revealing the signer's identity. Only the group trust authority can trace the signer's identity and revoke the group keys.

III. NETWORK SCENARIO

Here, we present our adversaries and attack models as well as the network assumptions and the node model.

A. Adversaries and Attack Models

Without loss of generality, we assume that an adversary knows all the network protocols and functions. The attackers outside the network do not know the secret keys, but those inside the network may know the keys. We classify their attacks according to their behaviors (e.g., active or passive) and locations (e.g., inside or outside the network).

Passive outside attack: There may be an external global passive adversary, who can observe and record all the wireless communications in the network. It will try to reveal the identities of the source, destination, and en-route nodes of a particular flow, or infer the traffic flows by linking the packets to the source or destination nodes.

Active outside attack: The passive attackers avoid any attack that reveals their actions since they attempt to be invisible, but the active outside attackers do not have such restrictions. They may aim to disrupt the routing or launch a DoS attack. They can move from here to there and launch attacks randomly.

Passive inside attack: The attackers are legitimate MANET nodes. Similar to the passive outside attackers, they will try to infer the identities of the source, destination, or enroute nodes without exposing themselves. Since they can read the legitimate packets, the traffic pattern or node mobility information may be learned by them.

Active inside attack: They can modify, inject, and replay genuine messages. They can also masquerade as other nodes and launch the impersonation attacks. They can create one or more phantom nodes by generating valid routing packets.

B. Network Assumptions

We denote a MANET by **T** and make the following assumptions.

1) Public Key Infrastructure: Each node **T** initially has a pair of public/private keys issued by a public key infrastructure (PKI) or other certificate authority (CA). For node ($A \in \mathbf{T}$), its public/private keys are denoted by KA+ and KA . Similar to the existing secure routing [7], we assume that there exists a dynamic key management scheme in **T**, which enables the network to run without online PKI or CA services.

2) Group Signature: We consider the entire network T a group and each node has a pair of group public/private keys issued by the group manager. The group public key, which is denoted by GT +, is the same for all the nodes in T, whereas the group private key, which is denoted by GA− (for $A \in T$), is different for each node. Node A may sign a message with its private key GA−, and this message can be decrypted via the public key GT + by the other nodes in T, which keeps the anonymity of A [11]. We also assume that there exists a dynamic key management scheme working together with the admission control function of the network, which enables the group signature mechanism running properly. Such assumptions are also adopted in the existing work of military ad hoc networks [2], [7].

3) Neighborhood Symmetric Key: Any two nodes in a neighborhood can establish a security association and create a symmetric key with their public/private keys. This association can be triggered either by a periodical HELLO messages or by the routing discovery RREQ messages. For two nodes A and B ($A, B \in T$), the shared symmetric key is denoted by KAB and used for the data transmissions between them. There are some approaches supporting the establishment of a one-hop shared key, such as MASK, RAODR, and USOR. In this paper, we

assume that one of the approaches is available in T. The notations are summarized in Table I.

C. Node Model

1) Destination Table: We assume that a source node knows all its possible destination nodes. The destination information, including one of destination's pseudonym, public key, and the predetermined trapdoor string dest will be stored in the destination table. Once a session to the destination is established, the shared symmetric key is required for data encryptions in the session. Such a

TABLE I
NOTATIONS FOR SECURITY PRIMITIVES

Notations	Descriptions
K_{A+}	Public key of node A
K_{A-}	Private key of node A
G_{T+}	Group public key of network **T**
G_{A-}	Group private key of node A
K_{AB}	Symmetric key shared by nodes A and B
$\{d\}K_{A+}$	Data d is encrypted by key K_{A+}
$[d]K_{A-}$	Data d is signed by node A
$\langle d\rangle K_{AB}$	Data d is encrypted by shared key K_{AB}
$(d)K_A$	Data d is encrypted by one symm. key of A
$O_K(m)$	Encrypted onion for message m with key K
N_A	One-time Nym. generated by A to indicate itself
$dest$	A special bit-string tag denoting the destination

symmetric key is generated by the source node before sending the route requests and stored in the destination table after receiving the route reply. For example, a sample entry of the destination table is (Dest_Nym, Dest_String, Dest_Public_Key, Session_Key).

2) Neighborhood Table: We assume that every node locally exchanges information with its neighbors. It can generate different pseudonyms to communicate with different neighbors. topology. The neighbor security associations are established as well as the shared symmetric keys. The information is stored in a neighborhood table. For example, a sample entry of the neighborhood table is (Neighbor_Nym, Session_Key).

3) Routing Table: When a node generates or forwards a route request, a new entry will be created in its routing table, which stores the request's pseudonym and the secret verifi-cation message in this route discovery. Such an entry will be marked in the status of "pending." If an RREP packet is received and verified, the corresponding entry in the routing table will be updated with the anonymous next hop and the status of "active." Meanwhile, a new entry will be created in the node's forwarding table. For example, a sample entry of the routing table is (Req_Nym, Dest_Nym, V er_Msg, Next_hop_Nym, Status). Note that, to simplify the notation, we ignore the timestamp information of the entry in the table.

4) Forwarding Table: The forwarding table records the switching information of an established route. We adopt the per-hop pseudonym as the identifier for packet switching, similar to the virtual channel identifier (VCI) in ATM networks. In each entry of the forwarding table, the route pseudonym is generated by the destination node, whereas the node pseudonyms of the previous and next hop are obtained after processing the related RREQ and RREP packets. For example, a sample entry of the forwarding table is (Rt_Nym, P rev_hop_Nym, Next_hop_Nym).

IV. PROTOCOL DESIGN

In this section, we present the design of OA2R protocol. Considering the nodal mobility, we take the on-demand ad hoc routing as the base of our protocol, including the phases of route discovery, data transmission, and route maintenance. In the route discovery phase, the source node broadcasts an RREQ packet to every node in the network. If the destination node receives the RREQ to itself, it will reply an RREP packet back along the incoming path of the RREQ. In order to protect the anonymity when exchanging the route information, we redesign the packet formats of the RREQ and RREP, and modify the related processes. As an example, we use a five-node network to illustrate the authenticated anonymous routing processes. The network is shown in Fig.1, in which the source node S discovers a route to the destination node D.

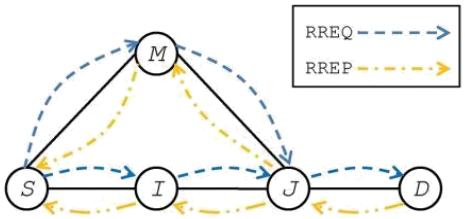

Fig. 1. Network topology

A. Anonymous Route Request

1) Source Node: We assume that S initially knows the information about D, including its pseudonym, public key, and destination string. The shortest path is computed to Destination D using Dijikstra,s. The destination string dest is a binary string, which means

"You are the destination" and can be recognized by D. If there is no session key, S will generate a new session key KSD for the association between S and D. The following entry will be updated in S's destination table.

Dest.Nym.	Dest.Str	Dest. Pub_Key	Session_Key
N_D	$dest$	K_{D+}	K_{SD}

Then, S will assemble and broadcast an RREQ packet in the format of (1). To simplify the notation, we ignore the timestamp information in the RREQ packet.

$$S \rightarrow *: [RREQ, N_{sq}, V_D, V_{SD}, Onion(S)]G_{S-} \quad (1)$$

where RREQ is the packet type identifier; Nsq is a sequence number randomly generated by S for this route request; VD is an encrypted message for the request validation at the destination node; VSD is an encrypted message for the route validation at the intermediate nodes; Onion(S) is a key encrypted onion created by S. The whole RREQ packet is finally signed by S with its group private key GS .

The combination of VD and VSD works similarly to the global trapdoor used in ANODR. We introduce VSD:

$$V_{SD} = (N_v)K_v \quad (2)$$

where Nv and Kv are two parameters created by S and sent to D for future route verification; Nv is a one-time nonce for the route discovery; and Kv is a symmetric key. The secret message VD is defined as:

$$V_D = \langle N_v, K_v, dest \rangle K_{SD}, \{K_{SD}\}K_{D+} \quad (3)$$

If D is the receiver of the message, D can decrypt the second part of VD by its private key KD , and then decrypt the first part by the obtained KSD. Otherwise, the receiver knows that it is not the intended destination. If S and D have already established KSD in a previous communication, the costly public encryption in the second part of VD can be eliminated, and then VD is defined as:

$$V_D = \langle N_v, K_v, dest \rangle K_{SD}, pad \quad (4)$$

where pad is a pre-defined bit-string that pads the message to a constant length. VSD and VD are separated in the RREQ format (1). For a non-destination node, it can use VSD as a unique identity for the route request. Now we describe the encrypted onion Onion(S). S creates the onion core as follow:

$$Onion(S) = O_{K_v}(N_S) \quad (5)$$

where NS is a one-time nonce generated by S to indicate itself. The core is encrypted with the symmetric key of Kv, and can only be decrypted by D via Kv.

After sending the RREQ, S creates a new entry in its routing table, which looks like the following:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N_D	V_{SD}	N/A	Pending

2) Intermediate Node: The RREQ packet from S is flooded in **T**. Now we focus on an intermediate node I, as shown in Fig. 1. We assume that I has already established the neighbor relationship with S and J. I knows where the RREQ packet comes from. The following entries are stored in I's neighborhood table:

Neigh. Nym.	Session_Key
N_S	K_{SI}
N_J	K_{IJ}

Once I receives the RREQ packet, it will verify the packet with its group public key GT+. As long as the packet is signed by a valid node, I can obtain the packet information. Otherwise, such an RREQ packet will be marked as malicious and dropped.

I checks the Nsq and the timestamp in order to determine whether the packet has been processed before or not. If the Nsq is not known in the routing table, it is a new RREQ request;if the Nsq exists in the table but with an old timestamp, it has been processed before and will be ignored; if the Nsq exists with a fresh timestamp, then the RREQ is a repeated request and will be recognized as an attack.

Then I tries to decrypt the part of VD with its own private key. In case of decryption failure, I understands that it is not the destination of the RREQ. I will assemble and broadcast another RREQ packet in the following format:

$$I \rightarrow *: [RREQ, N_{sq}, V_D, V_{SD}, Onion(I)]G_{I-} \quad (6)$$

where Nsq , VD, and VSD are kept the same as the received RREQ packet; the key-encrypted onion part is updated to Onion(I). The complete packet is signed by I with its group private key GI . I updates the onion in the following way:

$$Onion(I) = O_{K_{SI}}(N_I, Onion(S)) \quad (7)$$

where NI is a one-time nonce generated by I to indicate itself; Onion(S) is obtained from the received RREQ packet; this layer of onion is encrypted with the symmetric key KSI . When I's RREQ reaches the next hop J, J will perform the same procedures and update the onion in the RREQ with one more layer, which is:

$$Onion(J) = O_{K_{IJ}}(N_J, Onion(I)) \quad (8)$$

The routing tables of I and J will also be updated with a new entry as follow:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N/A	V_{SD}	N/A	Pending

3) Destination Node: When the RREQ packet reaches D,

D validates it similarly to the intermediate nodes I or J. Since D can decrypt the part of VD, it understands that it is the destination of the RREQ. D can obtain the session key KSD,

the validation nonce Nv, and the validation key Kv. Then D is ready to assemble an RREP packet to reply the S's route request.

B. Anonymous Route Reply

1) Destination Node: When D receives the RREQ from its neighbor J, it will assemble an RREP packet and send it back to J. The format of the RREP packet is defined as follow:

$$D \rightarrow *: (RREP, N_{rt}, \langle K_v, Onion(J)\rangle K_{JD}) \quad (9)$$

where RREP is the packet type identifier; Nrt is the route pseudonym generated by D; Kv and Onion(J) are obtained from the original RREQ and encrypted by the shared key KJD. The intended receiver of the RREP is J. 2) Intermediate Node: We assume that J has already established a neighbor relationship with I, D, and M. The following entries are already in J's neighborhood table: Neigh. Nym. Session Key

Neigh. Nym.	Session_Key
N_D	K_{JD}
N_I	K_{IJ}
N_M	K_{MJ}

If J receives the RREP from D, J will navigate the shared keys in its neighborhood table, and try to use them to decrypt ⟨Kv;Onion(J)⟩KJD. In case of a successful decryption, J knows the RREP is valid and from ND, and J also obtains the validation key Kv. Then J continues to decrypt the onion part. J knows the next hop for the RREP is NI . Then J will verify the linkage of the received RREP with its stored RREQ. It tries to use the obtained Kv to decrypt the

verification message VSD stored in its routing table. Once J finds the matched VSD, it will update the corresponding routing entry as follows:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N/A	V_{SD}	N_D	Active

Since Nv in VSD is not issued by J, J is not the source of the RREQ, then it has to assemble another RREP and forward it. The format of J's RREP towards the previous hop I is defined as:

$$J \rightarrow * : (RREP, N_{rt}, \langle K_v, Onion(I) \rangle K_{IJ}) \quad (10)$$

where Nrt and Kv are obtained from the received RREP; Onion(I) is obtained by from the decrypted Onion(J); the shared key KIJ is obtained from J's neighborhood table. The intended receiver of the RREP is I.

When the RREP packet travels according to the layers on the onion, it will start at the destination node and move back to its previous node. Each time the intermediate node can associate a value with the underlying wireless link on which the RREP travels, until the RREP packet reaches the source. In our protocol, every node records the one-time link pseudonyms announced by its neighbor node. Then the intermediate nodes' forwarding tables can be established after the RREP's trip. Now we discuss the forwarding table in detail. After J updates its routing table, it will also create a new entry in its forwarding table. It may record the multiple paths found in

the route discovery. According to the topology in Fig. 1, J's forwarding table may look like the following, in which NX;I stands for the ith one-time pseudonyms issued by node X: D issues different pseudonyms ND;1 and ND;2 to J. There are two forwarding relationships at J. NI;1 : ND;1 and NM;1

:

Rt. Nym.	Pre_hop Nym.	Next_hop Nym.
$N_{rt,1}$	$N_{I,1}$	$N_{D,1}$
$N_{rt,2}$	$N_{M,1}$	$N_{D,2}$

ND;2 describe the two routes of I − J −D and M − J −D,

as shown in Fig. 1. It can be seen that the forwarding table is made anonymous to any nodes, except for **the** switching node that owns the table. At the time of being anonymized, the switching relationship at each node en route can also be guaranteed.

intermediate nodes. If the decrypted onion core NS equals to one of S's

issued nonce, S is the original RREQ source. S will update its routing table as follow:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N_D	V_{SD}	N_I	Active

Then the route discovery process ends successfully. S is ready to transmit a data along the route indicated by **Nrt.**

C. Anonymous Data Transmission

Now S can transmit the data to D. The format of the data packet is defined as follows:

$$S \rightarrow D : (DATA, N_{rt}, \langle P_{data} \rangle K_{SD}) \quad (11)$$

where DATA is the packet type; Nrt is the route pseudonym that can be recognized by downstream nodes; the data payload is denoted by Pdata, which is encrypted by the session key KSD.

Upon receiving a data packet, every node will look into

its forwarding table. If Nrt in the data packet matches one entry in forwarding table, the node will forward the packet

to the anonymous next hop. Otherwise, the data packet will be discarded. Following the similar mechanism as the VCI in ATM network, the data packet can be switched along the route until it arrives at the destination.

D. Routing Procedure

The routing algorithm can be implemented based on the existing on-demand ad hoc routing protocol like AODV or

DSR. The main routing procedures can be summarized as follows:

1) During route discovery, a source node broadcasts an RREQ packet in the format of (1).

2) If an intermediate node receives the RREQ packet, it verifies the RREQ by using its group public key, and adds one layer on top of the key-encrypted onion, as (7). This process is repeated until the RREQ packet reaches the destination or expired.

3) Once the RREQ is received and verified by the destination node, the destination node assembles an RREP packet in the format of (9), and broadcasts it back to the source node.

4) On the reverse path back to the source, each intermediate node validates the RREP packet of (2) and updates its routing and forwarding tables. Then it removes one layer on the top of the key-encrypted onion, and continues broadcasting the updated RREP in the format of (10).

5) When the source node receives the RREP packet, it verifies the packet, and updates its routing and forwarding tables. The route discovery phase is completed.

6) The source node starts data transmissions in the established

route in the format of (11). Every intermediate node forwards the data packets by using the route pseudonym.

V.CONCLUSION

In this paper, we have designed an authenticated and anonymous routing protocol for MANETs in adversarial environments. The key-encrypted onion routing with a route secret verification message is designed to not only record the anonymous routes but also prevent the intermediate nodes from inferring the real destination. Compared to ANODR, AASR provides higher throughput and lower packets loss ratio in different mobile scenarios in the presence of adversary attacks.

The future work on OA2R will focus on malicious node detection and rejection mechanism, we will also look toward reducing packet delay and link maintenance by adding routing error phase.

ACKNOWLEDGEMENTS

The authors would like to whole heartedly thank the Management, principal staff and students K.S. Institute of Technology, Bangalore

REFERENCES

[1] Wei Liu and Ming Yu, “AASR: Authenticated Anonymous Secure Routing for MANETs in Adversarial Environments,” IEEE Trans. on Vehicular Technology,Volume:PP,Issue:99,Date of Publication :21.March.2014.

[2] R. Song and L. Korba, “A robust anonymous ad hoc on-demand routing,” in Proc. IEEE MILCOM,
Oct. 2009, pp. 1–7.

[3] M. Yu, M. C. Zhou, and W. Su, “A secure routing protocol against Byzantine attacks for
MANETs in adversarial environment,” IEEE Trans.

Veh. Technol., vol. 58, no. 1, pp. 449–460, Jan. 2009.

[4] S. Seys and B. Preneel. ARM : Anonymous routing protocol for mobile ad hoc networks. AINA Workshops, pages 133– 137, 2006. [13] D. Sy, R. Chen, and L. Bao. ODAR: On-demand anonymous routing in ad hoc networks. MASS, Oct 2006

[5] S. William and W. Stallings, Cryptography and Network Security, 4th ed. Delhi, India: Pearson Education India, 2006.

[6] L. Yang, M. Jakobsson, and S. Wetzel. Discount anonymous on de-mand routing for mobile ad hoc networks. SECURECOMM, 2006.

[7] X. Hong, J. Kong, Q. Zheng, N. Hu, and P. Bradford, “A hierarchical anonymous routing scheme for mobile ad-hoc networks,” in Proc. IEEE MILCOM, Oct. 2006, pp. 1–7.

[8] Y. Cheng and D. Agrawal. Distributed anonymous secure routing protocol in wireless mobile ad hoc networks. OPNETWORK, Aug 2005.

[9] D. Boneh, X. Boyen, and H. Shacham, “Short group signatures,” in Proc. CRYPTO, Aug. 2004, pp.
41–55.

[10] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba. SDAR:a secure distributed anonymous routing protocol for wireless and mobile ad hoc networks. LCN, 2004.
Network scenerio.

11] D. Boneh, X. Boyen, and H. Shacham, “Short group signatures,” in Proc. CRYPTO, Aug. 2004, pp. 41–55.

[12] J. Kong and X. Hong. ANODR: Anonymous on demand routing with untraceable routes for mobile ad-hoc networks. In MOBIHOC, 2003.
[13] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. Proceedings of the 13th USENIX Security Symposium, Aug 2004. [9] D. Goldschlag, M. Reed, and P. Syverson. Onion routing for anonymous and private internet connections. Communications of the ACM, 42, 1999.

BIOGRAPHIES

CHAITHRA S, Student of K.S Institute of Technology, Bangalore, Karnataka, India. Currently pursuing 8th semester in Computer Science and Engineering.

CHAITRA V, Student of K.S Institute of Technology, Bangalore, Karnataka, India. Currently pursuing 8th semester in Computer Science and Engineering.

ROOPESH KUMAR B N, Assistant Professor in Department of CSE, of K.S Institute of Technology, Bangalore, Karnataka, India.

[14] M. Reed, P. Syverson, and D. Goldschlag. Anonymous connections and onion routing. IEEE Journal of Selected Areas in Communications,, 16(4):482–494, May 1998.
[15] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Anonymous connections and onion routing," IEEE J. Sel. Areas Commun., vol. 16, no. 4, pp. 482–494, May 1998.