TRUST FRAMEWORK FOR DATA FORWARDING IN OPPORTUNISTIC NETWORKS USING MOBILE TRACES

B.Poonguzharselvi[1] and V.Vetriselvi[2]

[1,2]Department of Computer Science and Engineering, College of Engineering Guindy, Anna University Chennai, Chennai-600025.
biselvi@gmail.com
vetri@annauniv.edu

ABSTRACT

Opportunistic networks are usually formed spontaneously by mobile devices equipped with short range wireless communication interfaces. The idea is that an end-to-end connection may never be present. Designing and implementing a routing protocol to support both service discovery and delivery in such kinds of networks is a challenging problem on account of frequent disconnections and topology changes. In this network one of the most important issues relies on the selection of the best intermediate node to forward the messages towards the destination. This paper presents a trust framework for opportunistic network where the nodes in the network follow the trace based mobility model. The selection of next hop to forward the data packets is based on the trust value as well as the direction of movement of node towards the destination. The trust value is obtained from the trust framework of the data forwarding node. The direction of destination is obtained from the movement trace file that is maintained by the nodes in the network. In this proposed framework, the message is encrypted to secure both the data and path information. The effectiveness of this proposed framework is shown using simulation.

KEYWORDS

Opportunistic network, mobile trace, routing protocol, delay tolerant network

1. INTRODUCTION

Opportunistic network is a type of Delay tolerant network (DTNs), where data will be routed with tolerant delay from source to destination [7]. This type of network is used for emergency applications. An opportunistic network (oppnet) is a network of mobile as well as fixed nodes. Opportunistic network are different from traditional network. In traditional network, nodes are deployed together and end to end path exist for data forwarding. But in opportunistic network there is no fixed path between source and destination due to mobile nodes. It is an extension of mobile ad hoc network.

In opportunistic network [17], first seed oppnet is deployed like typical ad hoc network. Then it detects foreign devices for completing the applications with tolerant delay. The detected devices are evaluated for their usability and resource availability. If it is satisfied the device is integrated into the oppnet as a helper. This helper integration process is continued until enough devices are found for completing the task. Once the integration process is completed, routing will be done with the help of helper nodes. Helper nodes are used as intermediate nodes for forwarding data from source to destination. Different nodes make collaboration to exchange data from source to destination.

DOI : 10.5121/ijwmn.2012.4609

In this type of network, a data holding node (source node/neighbour node) finds an opportunity to forward data. When there is no direct connection between source and destination, data holding node will discover its nearest neighbour node and uses it to forward messages toward the destination node. Message is delivered hop by hop closer to the destination. In opportunistic network the routing is done with the help of intermediate helper nodes to deliver the data packets to destination. The applications of opportunistic network are typically used in an environment that is tolerant of long delay and high error rate.

The researchers have found the problem of selfishness in opportunistic networks. For example if a large set of nodes rely on other nodes to forward their messages, but are not willing to relay messages themselves, the routing network quickly breaks down. An opportunistic network is designed to forward messages, with the constraint set by the message originator that they should only be given to entities with a trust value above a certain threshold. This helps sender to forward message to destination through the trusted nodes to avoid selfish nodes involving in data forwarding. Since the selfish nodes may not forward data to destination thus it reduces the performance of the routing in opportunistic networks. This problem is the motivation for proposing a trust mechanism to avoid selfish users for message forwarding.

The computational trust has been selected as one of the solutions for the security problem in opportunistic network. A computational trust value is assumed to be very similar to the human notion of trust. Before a node in a network agrees to interact with another, it gathers information about the other node, and determines a trust value depending on previous interactions with this particular device or class of devices, on reputation values provided by third devices, and other application dependent data. This trust value can be applied to access control problems (is the device trusted enough) and to ensure confidentiality. Trust is not applicable as the only security measure in high security systems, but it gives good results. However, the application of trust values presents a unique challenge in the case of opportunistic network.

This paper is organized as follows. Section 2 focuses on related work; it contains various proposed methods for security mechanisms in opportunistic networks. Section 3 focuses on System architecture for proposed method. In Section 4 the implementation details are discussed and performance has been evaluated. Section 5 concludes the paper.

2. RELATED WORK

There are only few works have been done on security mechanisms for opportunistic networks. Shikfa et al [10] pointed out that the key management scheme in opportunistic networks should be non-interactive and nodes should not need prior contact with the destination to query the related public key. This is because end-to-end connectivity cannot be obtained. He proposed a novel security scheme by combining the identity-based cryptography and policy-based encryption to ensure the privacy of opportunistic routing protocols. However, in this scheme, there is a Trusted Third Party (TTP) that provides the key generation server for all the nodes. But it is hard to realize in opportunistic networks.

Shikfa et al [11] also proposed another key management which integrates the security associations with the neighbourhood discovery together by using the certificates and signature chains. Whereas, Shikfa et al still assumes that there is a single Identity Manager (IM) in opportunistic networks which generates and links the pseudonyms to mobile nodes for protecting the system from the Sybil attacks. Besides that, they assumed that the shared keys can be negotiated by sending a security association reply through the reverse path of the request. Both of the two assumptions which mentioned above are difficult to achieve.

Trifunovic et al. [12] considered that in a totally distributed and decentralized environment, traditional cryptography schemes become useless, since no authority servers are deployed. Moreover, the cryptography methods can only ensure the inimitability of users' identities, but no assured binding between real entity and identity can be obtained. Therefore, the establishment of trust can validate the nodes' identities and resist the Sybil nodes more effectively in opportunistic networks.

Ing Ray Chen [4] has proposed a trust mechanism by combining social and QOS related attributes. This protocol is resilient against bad-mouthing, good-mouthing and whitewashing attacks performed by malicious nodes. They considered connectivity to measure the QoS trust level of a node and also unselfishness and honesty to measure the social trust level of a node. Trust values are calculated and arranged with differnet weights. Each node is associated with this calculated trust value. This mechanism is tested on PROPHET routing algorithm. It shows that, the performance of the routing algorithm is improved by including this trust mechanism to avoid malicious nodes in data forwarding.

The privacy issues are the main problems in opportunistic networks, since the opportunistic routing protocols utilize the context information which may be sensitive to users. HiBOp protocol [13] enforces the privacy by implementing the concept of community. It assumes that exposing the social context information to the nodes which are in the same community is security, for nodes trust each other in the same community. Consequently, the messages can only be forwarded to the intermediate nodes which come from the same community with destination nodes. Obviously such a security scheme will bring a bad influence on the routing performance, for only limited nodes can be choose as the candidate forwarders.

This paper proposes trust based data forwarding in opportunistic networks, where the data forwarding follows a trace based routing algorithm. The forwarding of data depends on trust as well as direction of the next hop, whether it is moving in the direction of the destination. It provides security and also the reach ability of the destination.

3. TRUST FRAMEWORK

Opportunistic network is composed of seed nodes and helper nodes. The proposed trust framework based data forwarding is composed of the following main functionalities,

3.1. Initial Trust Value Setup

We assume that the maximum trust value is one and the minimum is zero. So trust value for a node lies in [0, 1]. Initial trust value for seed nodes (source / destination) is assumed to be one, since they are fully trusted nodes. But for helper nodes, initial trust value has to be calculated. The initial trust value for helper node is based on the friendship vector associated with nodes in the network. The friendship vector for a particular node indicates a list of known other nodes to this node. Some helper nodes may not have friendship vector. It indicates that this node does not have any friends in the network. So the initial trust value for this node is assumed to be zero. For the remaining helper nodes, their initial trust value is calculated from their friendship tie in the network. The calculated trust value lies in the range of [0,1].

For a node N_i, its trust value is calculated by its friendship tie in the network. Suppose it has m friends in the network then its trust value is calculated using the following algorithm (initial trust value setup). In this algorithm, T_i denotes the trust value of a node N_i in the network. If N_i is a seed node then its trust value is 1. If it is not a seed node, the trust value is calculated from the friendship vectors of the seed nodes in the network. If N_i is a helper node and there are no

friends for this node then the trust value for this node is 0. The algorithm for calculating the initial trust value for the nodes in the network is shown in Figure 1.

Algorithm : Initial Trust Value Setup

Input : nodes, friendship vector

Output: nodes with trust value

N_i : a node in the network

S_i : seed node in the network

T_i : trust value of node N_i

FR_i : friendship vector of a node N_i

Step 1: If node N_i is a seed node then trust value of N_i= 1

(ie) $T_i(N_i)= 1$

Step2: For a helper node N_i, search for N_i in other node's friendship vector FR_j.

Step3: Count the number of friendship vectors FR_j where N_i appears.

Step4: Take the trust values of the nodes whose friendship vectors are selected in step2.

Step5: Calculate the trust value for a helper node N_i using the following formula,

$$T_i(N_i)= \{\Sigma^{n}_{j=0} T_j(N_j)\}/ n ;$$

Figure 1. Initial Trust Value Calculation Algorithm

3.2. Trust Table Creation

Trust table is created and maintained for all the nodes in the network. After calculating the initial trust value, the trust table is created to store the trust values of known nodes.

Trust Table(M_i)

Node ID	Trust Value
N_j	x
-	
N_s	y

Figure 2. Trust Table for Node M_i in the Network

Trust table is created with node ID of the known node and its corresponding trust value as shown in fig 2. It shows the trust table created for the node N_i in the network. It stores the known node's ID and its corresponding trust value. This information is used while selecting the trusted next hop forwarder for forwarding data to destination. This trust table is updated at regular interval of time.

3.3. Data Forwarding

Data forwarding in opportunistic network follows "Store and Forward" mechanism. Since there is no end to end path between source and destination, the data to the destination can be forwarded through the intermediate nodes. It is assumed that the seed nodes have the trace file information of the other seed nodes. Hence the source node knows the probable position of the destination from the trace file of the destination. The major task is the selection of next hop forwarder [18].

Once the trace file of the destination is available with the source node, the movement of the destination node can be predicted by looking up its trace file. Instead of broadcasting a data in all directions, the source node and the intermediate nodes can use directional information to forward data to destination. This helps to reduce the load on the network. An additional field containing the destination's probable position can be added in the message header.

Here the selection of trusted next hop forwarder places a major role. Since if the data is forwarded to malicious node, it can either drop or forward that data with long delay. So the source node has to select the trusted next hop forwarder for forwarding data to destination. First the source node creates message for the destination. The data packet format for this message is shown in Figure 3.

<table>
<tr><td>Source address</td><td>Destination address</td><td>Packet ID</td><td>Packet size</td><td>TTL</td><td>Destination position</td><td>Next hop address</td></tr>
<tr><td colspan="7">Data</td></tr>
<tr><td colspan="7">Authentication message</td></tr>
</table>

Figure 3. Data Packet Format

The additional fields in this packet (highlighted in the figure), when compared to any routing protocol, are the destination position, authentication message and the next hop address. The data is forwarded to the next hop that moves in the direction of the destination. Using the trace file of the neighbour nodes the next hop node to forward the data is identified. The authentication message field indicates the encrypted form of path information. The path information is encrypted using encryption algorithm. This authentication message can be accessed only by the seed nodes in the network, since authentication message is in encrypted form. So the helper node cannot view or modify this message. After the source node creates the message for the destination, the following processes are carried out to forward that message to the destination.

3.3.1 Data Encryption

Source and destination nodes are having a common shared secret key for data encryption and decryption. This shared secret key is known only to the seed nodes in the network. Helper nodes do not know the secret key. So they will not be able to encrypt or decrypt the data.

When a source node wants to forward a data to destination, first it encrypts that data with common shared secret key. Then it forwards data to destination through the intermediate helper nodes. The intermediate helper nodes cannot decrypt the data or view the message, since they do not know the secret key for data decryption. Instead of forwarding data to all neighbour nodes, data is forwarded to the nodes which are going in the direction of destination. Hence the network congestions can be avoided.

3.3.2 Trusted Next hop Selection

This process is continued until the data reaches the destination. Before trusted next hop selection, the neighbour nodes should be identified. Neighbour node identification is the process whereby a node identifies its current neighbours within its transmission range. For a particular node, any other node that is within its radio transmission range is called a neighbour. All nodes consist of neighbour set which holds details of its neighbour nodes. Since all nodes might be moving, the neighbours for a particular mobile node are always changing.

The data holding node (source/ intermediate node) identifies its current neighbours. The neighbour set is dynamic and needs to be updated frequently. Generally, neighbour node identification is realized by using periodic beacon messages. The beacon message consists of node ID, node location and timestamp. Each node informs other nodes of its existence by sending out beacon message periodically. Inform all nodes within the transmission range of source/packet forwarding node to intimate its presence by sending a beacon message every particular seconds.

From the identified set of neighbour nodes, the data holding node selects the node which is going in the direction of the destination node. Once the next hop node is selected, trusted node has to be selected. Since there may be some malicious nodes, that can drop or forward the data with long delivery delay. A node is a trusted next hop node, if it satisfies the following condition.

Condition: Trust (node) >= Trust Threshold ------- (1)

If a next hop node satisfies the above condition, then that node will be selected as a trusted next hop forwarder. In equation (1), trust threshold indicates the user selected trust limit for selecting the trusted node for data forwarding. According to the type of the application, trust threshold varies. After selecting the trusted next hop forwarder, authentication message process is started.

3.3.3. Authentication Message

The authentication message (AM) consists of intermediate node ID's (i.e the helper node ID's) that involves in forwarding the data towards the destination, it will be in an encrypted form. This message is created only by the seed nodes that share a common secret key. Conventional encryption technique is followed for encrypting because of the cost and speed limitations in the mobile nodes. This process is started after the selection of the trusted next hop forwarder. In order to include the trusted next hop node, the authentication message has to be decrypted for decrypting this, shared secret key is needed. Only seed nodes can access the message and add IDs into it.

For seed nodes, it is easy to add the trusted next hop forwarder ID into the AM field. But for the helper nodes, it is very difficult to add the trusted next hop forwarder ID into the AM, since they do not know the shared secret key. The AM process is shown in Figure 4.

To access the AM, the shared secret key is must. When a helper node wants to handover the data to another helper, it gets the help of a seed node to include the AM. Therefore if the intermediate node is a helper node, it requests the nearby seed nodes to add the selected next hop node into the AM.

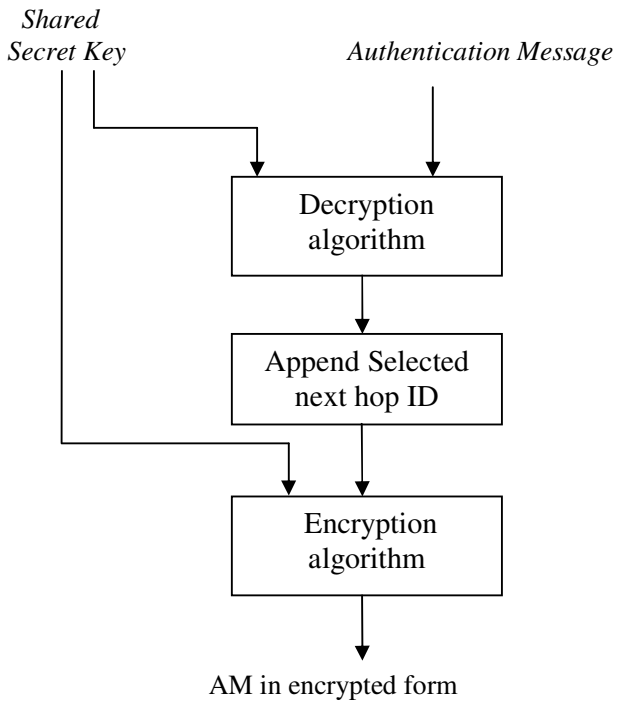

Figure 4. AM Inclusion by Seed Node

The assumption we make here is that a seed node is present when a helper node forwards the data to another helper. The seed node will add the next forwarding node ID into the authentication message. So the seed node decrypts the AM using decryption algorithm with shared secret key. Then adds the given node Id into the AM and encrypts the AM using encryption algorithm. Finally this seed node forwards the message to the requested helper node. After receiving this message, the helper node forwards the data packet to the selected next hop node. This process is repeated until the data reaches the destination.

3.3.4. Data Decryption

Once the data reaches the destination node, it decrypts the encrypted message using the common shared secret key and also it decrypts the AM which is appended with the data. Both AM and data decryption is done using decryption algorithm, since this common secret key is known to all the seed nodes. So the helper nodes cannot decrypt the data or AM. Trust table update process is started after decrypting the received data.

3.3.5. Updating Trust Table

The trust table is updated at the destination node whenever it receives the data. Figure 5 shows the process of updating the trust table by the destination. After decrypting the data and AM, the

destination node looks into the nodes in the AM. Then the destination node verifies the received data and checks for its correctness. If the received data contains no error then, the destination increases the trust values of the nodes in the AM. If the received data contains error, then the destination decreases the trust values of the nodes in the AM. For all the nodes in the AM, the trust value is updated in the destination node's trust table. After updating this trust table, it is distributed to other neighbour nodes in the network.

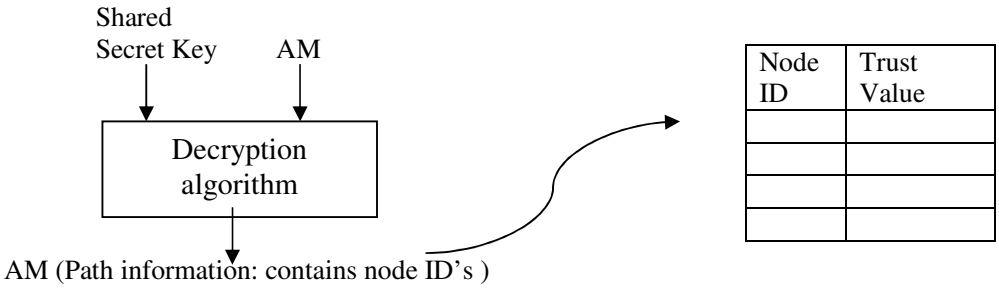

Figure 5. Updating Trust Table by the destination

Above discussion shows the forwarding of data from source to destination through the intermediate nodes. The presents of malicious nodes will not affect the data forwarding. The intermediate nodes are selected based on the trust value. So the node which has the required trust value will participate in the forwarding of the data from source to destination.

4. IMPLEMENTATION AND PERFORMANCE ANALYSIS

The proposed method is simulated using ONE simulator [9] which combines movement modelling, routing simulation, visualization and reporting in one program. It is agent-based discrete event simulation engine and provides interface for developing new routing algorithms and mobility models. New messages are generated by the Message_Event_Generator in ONE. The trace file incorporation is done with the help of External_Movement class in ONE simulator. The simulation parameters are shown in Table 1.

Table 1. Simulation parameters

Parameter	value
Simulation time (sec)	3000s
Simulation area	1000,1000
No. of nodes	50
Mobility Model	External Movement
Transmit range (meters)	6
Transmit Speed (m/s)	250

The following evaluation parameters are used to evaluate the proposed routing algorithm,

a. Varying Malicious Nodes
Changing the number of malicious nodes in the network.

b. Varying Trust Threshold
Changing the trust threshold value for selecting the trusted next hop forwarder.

c. Delivery Ratio
Delivery ratio is defined as the ratio of the number of successfully delivered messages to the number of all messages generated.

d. Overhead Ratio
Time taken to transmit data in the opportunistic network.

The proposed trust framework based data forwarding in opportunistic networks evaluated by varying the trust threshold value and the strength of the malicious node in the network.

4.1 Varying Malicious Nodes

The trust framework based data forwarding is evaluated by changing the number of malicious nodes in the network. In Figure 6, the performance of proposed method is shown by comparing without trust framework and with trust framework in mobile trace based routing algorithm. This comparison is done by varying the number of malicious nodes in the network. The delivery probability for the routing algorithm without trust framework decreases when the number of malicious nodes in the network increases.

The delivery probability of the routing algorithm with trust framework is high as compared with delivery probability of the routing algorithm without trust framework. This is because the data if forwarded with the helper nodes that satisfy some trust threshold value. But the presence of more number of malicious nodes in the network slightly affects the performance of the routing algorithm with trust framework.

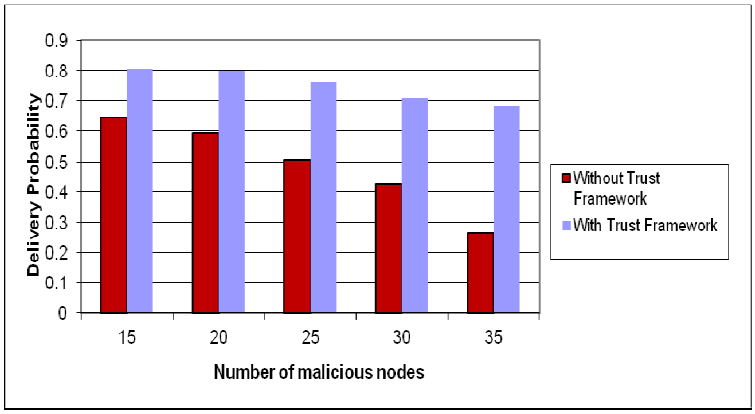

Figure 6. Comparisons by Varying Malicious Nodes

4.2 Varying Trust Threshold

The trusted next hop forwarder is selected based on the trust threshold value. So the selection trust threshold affects the performance of routing algorithm. The proposed trust framework based data forwarding is evaluated by changing the trust threshold value as shown in Figure 7 and 8.

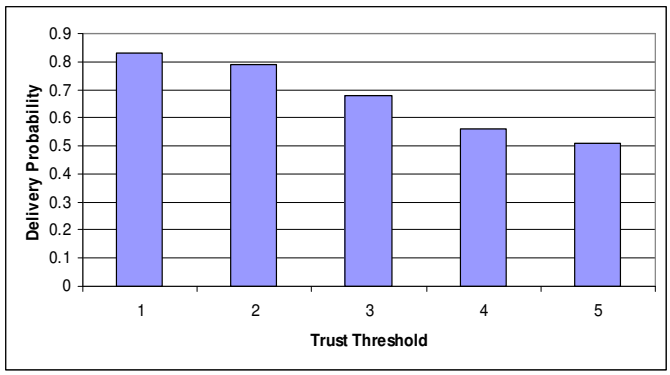

Figure 7. Delivery Ratio for Different Trust Threshold

From Figure 7, when the trust threshold is very low, the delivery probability is high. If the threshold value is increased, the delivery probability decreases. Since the number nodes selected for data forwarding is limited at high threshold value.

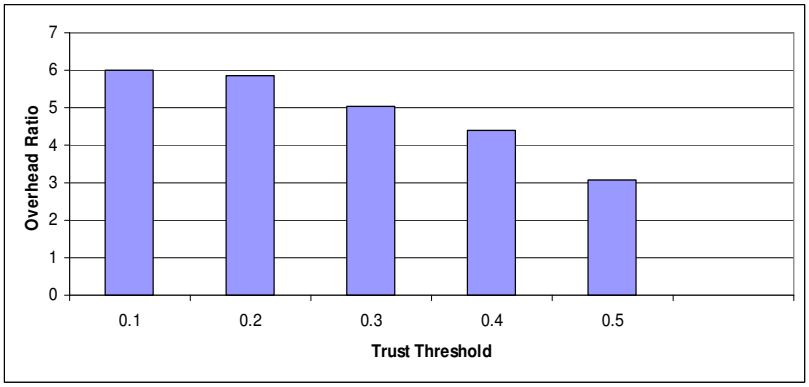

Figure 8. Overhead Ratio for different Trust Threshold

From Figure 8, overhead decreases with increase in trust threshold. Since when the trust threshold is low, more number of next hop nodes is selected for data forwarding towards destination. So the number of relayed messages increases. This leads to congestion in the network. But when the trust threshold is high, the overhead in the network is tolerable. It leads to reduced congestion in the network.

From the above discussions, the proposed trust framework based data forwarding performs better than the routing algorithm without trust framework. But the trust threshold has to be selected carefully. Since delivery probability and overhead ratio are decreasing with increase in trust threshold. So the selection of trust threshold is important. The trust threshold has to be selected in such a way that the delivery probability is high and overhead ratio is low.

5. CONCLUSION

Mobile devices with local wireless interfaces can be organized into opportunistic networks which have the properties of social networks. To take advantage of this attribute, we propose a trust framework based data forwarding in opportunistic networks which uses mobile traces for node movements to predict the time at which two nodes will be in contact, and therefore it improves the performance of message forwarding through trusted intermediate nodes. Also

trust will be assigned for each node in the network and the next hop forwarder is selected based on the trust threshold. Experimental result shows that proposed framework has high delivery probability. Thus setting appropriate trust threshold, better delivery probability can be obtained. Also the performance of the proposed trust framework is analysed with different number of malicious nodes in the network.

The future extension is to investigate other forms of attacks by malicious nodes such as jamming, forgery and Denial-of-Service (DoS) attacks. This can be achieved by considering other trust metrics such as technical competence, centrality and similarity.

REFERENCES

[1] A. Lindgren, A. Doria & O. Scheln,(2003) " Probabilistic Routing in Intermittently Connected Networks", ACM *Proceedings Newsletter on Service Assurance with Partial and Intermittent Resources*, Vol. 7, No.3, pp 19-20.

[2] A. Vahdat & D. Becker, (2000) "Epidemic Routing for Partially-Connected Ad Hoc Networks", Technical Report CS-200006, Duke University.

[3] B. Burns, O. Brock & B. N. Levine, (2005) "MV routing and capacity building in disruption tolerant networks", IEEE *Conference on Computer and Communications and Societies*, Vol. 1, pp. 398–408.

[4] Ing Ray Chen, Fenye Bao, MoonJeong Chang & Jin-Hee Cho, (2011) "Integrated Social and QoS Trust-Based Routing in DelayTolerant Networks". *In Wireless Personal Communications, Springer*, Vol. 66, pp. 443–459.

[5] J. Burgess, B. Gallagher, D. Jensen & B. N. Levine,(2006) "Maxprop:Routing for vehicle-based disruption-tolerant networks", IEEE *International Conference on Computer Communications*, pp. 1–11.

[6] N. Jianwei, G. Jinkai & C. Qingsong, (2011) "Predict and Spread: an Efficient Routing Algorithm for Opportunistic Networking", IEEE *Conference on Wireless Communications and Networking*, pp. 498–503.

[7] K. Fall, (2003) "A delay-tolerant network architecture for challenged Internets", ACM *Conference on SIGCOMM*, pp. 27-34.

[8] K.Prasanth, K.Duraiswamy, K.Jayasudha & C.Chandrasekar, (2009) "Edge Node Based Greedy Routing for VANET with Constant Bit Rate Packet Transmission". ACEEE *International Journal of Recent Trends in Engineering*, Vol. 2, No. 4.

[9] Keranen A, Ott J & Karkk aiene T, (2009) "The ONE simulator for DTN protocol evaluation", ICST *Proceedings of the 2nd International Conference on Simulation Tools*, pp. 1–10.

[10] Shikfa A, Onen M & Molva R, (2009) "Privacy in context-based and epidemic forwarding".*In Proceedings of the IEEE International Symposium on World of Wireless, Mobile and Multimedia Networks & Workshops*, pp. 1-7.

[11] Shikfa A, Onen M & Molva R, (2010) " Bootstrapping security associations in opportunistic networks". *In Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pp. 147-152.

[12] Trifunovic S, Legendre F & Anastasiades C, (2010) "Social Trust in Opportunistic Networks". *In proceedings of the INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1-6.

[13] Boldrini C, Conti M, Jacopini J & Passarella A, (2007) "HiBOp: a History Based Routing Protocol for Opportunistic Network". IEEE *International Symposium on World of Wireless, Mobile and Multimedia Networks,* pp. 1-12.

[14] T. Spyropoulos, K. Psounis & C. S. Raghavendra, (2004) "Single-copy routing in intermittently connected mobile networks", IEEE *Conference on Sensor and Ad Hoc Communications and Networks*, pp 235 – 244.

[15] T.Spyropoulos, Thrasyvoulos & Psounis, (2009) "Spray and Focus: Efficient Mobility Assisted Routing for Heterogeneous and Correlated Mobility", IEEE *International Conference on Pervasive Computing and Communications Workshops*, pp 79-85.

[16] T. Spyropoulos, K. Psounis & C. S. Raghavendra, (2005) "Spray and wait: Efficient routing in intermittently connected mobile networks", ACM *Proceedings of SIGCOMM workshop on Delay Tolerant Networking*, pp 252-259.

[17] L. Lilien, Z.H. Kamal, V. Bhuse & A. Gupta, (2007) "Opportunistic networks: The Concept and Research Challenges in Privacy and Security", Chapter in: Mobile and Wireless Network Security and Privacy, Springer Science & Business Media, Norwell.

[18] B.Poonguzharselvi & V.Vetriselvi, (2012) "Data forwarding in Opportunistic Network Using mobile traces", *International Conference on Information Technology Convergence and Services.*