SRSnF: A Strategy for Secured Routing in Spray and Focus Routing Protocol for DTN

Sujoy Saha[1], Rohit Verma[1], Satadal Sengupta[1], Vineet Mishra[2], and Subrata Nandi[1]

[1] Dept. of Computer Science & Engineering, National Institute of Technology, Durgapur
[2] Dept. of Computer Applications, National Institute of Technology, Durgapur, India–713209
{sujoy.ju,satadal.sengupta.nit,rohitverma.kgp,vineet0309, subrata.nandi}@gmail.com

Abstract. This paper deals with the aspect of security in Delay Tolerant Networks (DTN). DTNs are characterized with decentralized control. Network performance and trustworthiness of transmitted information in DTNs depend upon the level of co-operation among participating nodes. As a result, DTNs are vulnerable towards untoward activities arising out of node selfishness as well as malicious intentions. In this paper, we limit our focus to the *Black Hole Denial-of-Service* attack. We develop a table-based strategy to record network history and use this information to detect discrepancies in the behavior of nodes, followed by elimination of those detected as malicious. We explain our detection mechanism considering *Spray & Focus* routing protocol as the representative routing scheme. The detection mechanism has been described in detail with examples pertaining to various case scenarios. Furthermore, we study the effect of variation of various parameters on detection efficiency and message transmission through simulation results.

1 Introduction

Wireless ad hoc networks are able to perform message transmission without fixed network infrastructure. However, in practice, ad hoc routing protocols do not work efficiently due to high node mobility, low node density, and short radio ranges. To deal with such loopholes, the concept of Delay Tolerant Network [1] (DTN) was introduced. These types of networks use store-and-forward approach to deal with challenging networking scenarios, e.g., battlefield and deep-space communication.

In DTNs, the packet is stored in the buffer of a node if it does not find the next-hop node en route to the destination. The metric-based routing protocols in DTN, e.g. Spray & Focus [2], PRoPHET [3], MaxProp [4], firstly store the packet in memory; then transmit it to particular nodes based on some delivery metric, such as some utility value (determined using some pre-defined utility function).

However, in DTN, it is a challenging task to verify the integrity and trustworthiness of transmitted information, mainly due to network constraints like low power availability, low node density, etc. Therefore, DTNs are vulnerable to adversary attacks and internal compromises, taking advantage of which, insider attacks may be launched. Insider attacks can cause significant disturbances.

N. Meghanathan et al. (Eds.): Advances in Computing & Inform. Technology, AISC 176, pp. 159–169.
springerlink.com

In this paper, we propose a scheme which uses *Spray & Focus* [2] routing protocol. Spray & Focus exhibits better performance that other utility-based mechanisms in most practical applications, which is the reason for it being our choice. The strategy is to check each node by its peer node for maliciousness. Detection is based on entries in tables maintained at each node which get verified and updated during each encounter between 2 nodes. Each node is capable of such integrity verification. Like in *Spray & Focus*, utility values have been calculated using transitivity too. Detailed description of strategy followed by comprehensive evaluation through simulations using ONE [5] has been carried out.

The paper has been divided into 5 sections. Section-2 gives a brief description of related work in maliciousness detection. Section-3 describes in detail the proposed strategy and incorporates case scenarios and algorithms. In section-4, we evaluate performance of our strategy by presenting relevant simulation results. Finally, in section-5, we conclude and point out areas for further research.

2 Related Work

A large amount of work has been done in the field of security in ad hoc wireless networks. A large number of threats that have been identified in case of ad hoc networks are applicable in some form or another in DTNs as well; however, due to non-availability of an end-to-end path in case of DTNs, those either fail completely, or perform miserably in a DTN scenario. Therefore, security approaches which are tailored to suit DTN characteristics are required. Strategies that have been proposed till date can be broadly classified into 3 categories: *reactive approach* which tries to mitigate the effects of malicious activities by identifying their source and taking prohibitory action, *proactive* approach which provides incentives to co-operative nodes thus encouraging participating nodes to forward messages whenever possible, and *user-interest oriented approach* where rational behavior of users based on personal interests or social ties are considered as design constraints.

The schemes that follow a *reactive approach* are as follows. Chuah et. al. proposed a ferry based detection scheme (FBIDM) [6]. In this scheme, detection is carried out by trusted examiners, i.e., ferry nodes. This strategy was improved upon by Ren et. al. by considering the property of transitivity (shown to be an important attribute for calculating delivery probability) in their paper called MUTON [7]. MUTON, like FBIDM, also used ferries to detect malicious nodes; MUTON was shown to perform much better. Meanwhile, Li et. al. proposed a strategy with encounter tickets [8] to thwart blackhole attacks. The scheme required forwarding nodes to generate signed encounter tickets to build an encounter history, which was later interpreted appropriately to make forwarding decisions. Ren et. al. came up with a packet exchange recording scheme [9] later, which required each node to maintain 2 tables for storing records of packet exchange; each node was taken through a sanity check at every encounter, failing which the node was blacklisted.

In the *proactive approach*, Zhu et. al. proposed SMART [10], a secure, multi-layered coin based approach which provided incentives to forwarding nodes if a packet relayed by them reached the destination successfully. However, forwarding nodes were deprived of credit when successful delivery failed due to faulty action of another node or expiration of time-to-live. This problem was tackled by Lu et. al. in

Pi [11], which, in addition to providing credit for successful delivery, ensured a boost in reputation for each forwarding action by a node.

Rational nature of users is given priority in the *user-interest oriented approach*. Li et. al. proposed socially selfish routing [12] which ensured that nodes made forwarding decisions based on strength of their social ties with other nodes. Ning et. al. proposed an incentive aware DTN [13] where user interest, e.g. news, sports, or entertainment, was taken into consideration to ensure user satisfaction.

In [14], we present a concise description of the work done so far by us on this field and results achieved, as well as indications for future study.

3 SRSnF: Secured Routing for *Spray & Focus* Protocol

We propose a table based scheme for ensuring secured routing using *Spray & Focus* routing protocol. Our scheme follows the *reactive approach* described in the previous section. In our scheme, each node maintains a table with entries corresponding to each encounter in the network; this is called *Network Records Table* (NRT). Additionally, each node stores a *Malice Identification List* (MIL) with node IDs of detected nodes in the network. Whenever 2 nodes (say A & B) encounter each other, they perform the following actions, in the specified order:

1. Node A checks its *Malice Identification List* for node ID of B. Node B performs the same *MIL* check for A. If any of the two tests positive, no further interaction takes place. If both test negative, they move on to the next step.
2. Node A and node B exchange their respective *Network Records Tables*.
3. Node A performs a *Maliciousness Test* on node B by matching its own NRT entry by entry with that of node B. B performs the same test on A.
4. If one of the nodes (say, node A) finds the other node (node B) to have engaged in malicious behavior, then A appends the node ID of B to its MIL. No further interaction takes place under this circumstance.
5. If each node passes the *Maliciousness Test* performed by the other node, then they update their *Network Record Tables* by inserting entries corresponding to latest information about the network. So, entries in NRT of A which are not present in NRT of B, are inserted into NRT of B and vice-versa.
6. Also, the nodes exchange and update their MIL in the same way that they do in case of the NRTs in order to know about all detected nodes.
7. Finally, they will exchange messages according to *Spray & Focus* protocol; then generate and exchange corresponding entries to be stored in their NRTs.

For our scheme, we shall consider the *Black Hole Denial-of-Service* (DoS) attack, i.e., malicious nodes will drop all packets they receive. Each such node will participate in routing of the message but are not going to forward the packet any further. Our strategy consists of 3 phases: 1st phase states pre-requisites for the network; next 2 phases describe detection in the *spray* and *focus* phases of *Spray & Focus*.

A. Authentication Phase: Before routing starts in the network, it is essential for each participating node to be aware about all other participating nodes; otherwise, external intrusion would become rampant. There is an Authentication Authority (AA) to ensure this. It assigns a unique node ID and a pair of public and private keys to each

participating node. Also, AA ensures that each node is preloaded with the list of node IDs of all participating nodes, and corresponding public keys. We assume that either this list doesn't change throughout the working of the DTN, or AA is able to modify suitably settings of all nodes in case of changes.

Whenever a node generates an entry corresponding to an encounter with another node, it is signed and encrypted using its pair of public & private keys. Such an entry can be decrypted by another authenticated node using its set of keys. Securing information using such techniques is a different field altogether and we do not discuss such techniques in this paper; rather we state that by using such a technique, we can ensure that modification of the contents of any entry is rendered impossible, thus securing the network from information forging attacks.

Furthermore, let us assume that AA generates a unique sink node ID, which is never assigned to any participating node, but is included in the list loaded into all nodes. Whenever a node is forced to purge a message from its memory due to buffer constraints, it generates an entry with this sink node ID as the receiver node. During Maliciousness Test, the checking node counts all such entries and includes this figure in its evaluation to ensure that there is no false report.

B. Spray Maliciousness Detection Phase: In spray phase, maliciousness can be shown by accepting packets from neighboring nodes, then dropping them instantly. Let such behavior be known as Spray Maliciousness; nodes exhibiting it be known as spray malicious nodes. Let us consider an example with these events:

1. E_0: Node A generates 3 copies of a message with ID 7002#.
2. E_1: Node A encounters node B and transfers 1 token for the message to it.
3. E_2: Node A encounters node C and transfers 1 token for the message to it.

During E_1, A & B update their NRTs as in Tab.1. During E_2, A & C update their NRTs as in Tab.2. Note that only the sections of NRTs relevant to the Spray Phase have been shown in Tabs.1,2,3. Let us consider following possible events after E_2:

CASE-1: ***Node B encounters A or C before encountering other node***- If node B is malicious, it will drop message after E_1. Now it does not contain any copy of the message, although its NRT has an entry showing reception from A. Whenever B comes in contact with A or C and poses as a relay node, its NRT will be checked. It will be found that B has an entry from A but no message to show for it; hence B is malicious, and is blacklisted. If B is regular, no anomaly would be found.

CASE-2: ***Node B encounters node D before encountering A or C-*** If node B is regular, and D is a better relay node than B itself, it will pass its message copy on to D. Then B and D will update their NRTs as shown in Table 3. So, when A or C encounters it at a later stage, it will know from B's NRT that no malicious activity has been exhibited. If B is malicious, node D will be able to ascertain that B had received a message copy from A but no longer has it; hence D will blacklist it.

Table 1. Spray Phase section of Network Record Table with node A and node B after E_1

Sender	Receiver	Message ID	Time to live	Copies with sender	Copies with receiver
A	B	7002#	5	2	1

Table 2. Spray Phase section of Network Record Table with node A and node C after E_2

Sender	Receiver	Message ID	Time to live	Copies with sender	Copies with receiver
A	B	7002#	5	2	1
A	C	7002#	5	1	1

Table 3. Spray Phase section of NRT with node B & D if they exchange copy

Sender	Receiver	Message ID	Time to live	Copies with sender	Copies with receiver
A	B	7002#	5	2	1
B	D	7002#	5	0	1

Algorithm 1. Spray Phase Detection Algorithm

```
//Key: Checker = checking node, peer = checked node
BEGIN
If checker has encountered peer previously, then
              If time_to_live has not expired, then
                  If(no.of copies received – no.of copies delivered –
                  no.of entries to sink node) <> 0
                      If lack of sufficient contact time is not the rea-
                      son for message drop/s, then
                        Peer is malicious, append peer to MIL;
                      EndIf
                  EndIf
              EndIf
Else
              Follow Spray phase mechanism;
EndIf
END
```

C. Focus Maliciousness Detection Phase: In Spray & Focus, a utility value $\tau_i(j)$ is defined for each pair of nodes, which indicates the prob. of node i to deliver the message to node j. When a node has only 1 copy of the message left, it passes it on to a node with better utility for destination. We assign the utility values for each pair of different nodes initially as infinity. $\tau_i(j)$ increases by 1 at every clock tick.

Let us consider 2 nodes A and B. At the beginning, $\tau_A(B) = \infty$ and $\tau_B(A) = \infty$. When they encounter $\tau_A(B) = \tau_B(A) = 0$. As soon as the connection is lost, this value starts increasing by 1 at every clock tick. It is also updated by transitivity:

$$\text{If } (\tau_B(C)) < (\tau_A(C) - t_m(d_{AB})), \text{ then } \tau_A(C) = \tau_B(C) + t_m(d_{AB}) \quad (1)$$

where $t_m(d_{AB})$ is time to cover distance AB under given mobility model m. This can be evaluated through calculation of velocity of node movement using traces.

Maliciousness can be shown by nodes by declaring fake lesser utility values (less being better in our example) for the destination. Let such behavior be known as *Focus Maliciousness*; nodes behaving in such fashion be known as *focus malicious nodes*. If successful, such nodes will receive a large fraction of packets from regular nodes, and drop them. In our strategy, a record for every change in utility value of every node is stored. An encountering node can determine the actual utility value of the node from these records. If the declared utility value is less than the calculated one, then it is tagged as malicious by the encountering node.

Let us take an example and calculate utility values of nodes for every other node. Let there be 4 nodes - A, B, C, D. Let *starting time* denote time when first contact is disconnected. $t_m(d_{ij})$ is assumed as 8 throughout. Let the events be:

1. Connection between A and B is lost at the *starting time*.
2. After 15 time units connection between A and C is lost.
3. After 10 time units connection between C and D is lost.
4. After 20 time units connection between A and D is lost.
5. After 5 time units connection between B and D is lost.
6. After 7 times units connection between B and C is lost.

Final status of NRTs relevant to *Focus* will be as in Tables 4, 5, 6 respectively.

Table 4. Focus Phase section of Network Record Table with node A

Sender	Receiver	Time Stamp	Low Utility Value	High Utility Value
A	B	00	--	--
A	C	15	$\tau_A(B) / 15$	$\tau_C(B) / \infty$
C	D	25	$\tau_C(A) / 10, \tau_C(B) / \infty$	$\tau_D(A) / \infty, \tau_D(B) / \infty$
A	D	45	$\tau_D(C) / 20, \tau_A(B) / 45$	$\tau_A(C) / 30, \tau_D(B) / 61$

Table 5. Focus Phase section of Network Record Table with nodes B and C

Sender	Receiver	Time Stamp	Low Utility Value	High Utility Value
A	B	00	--	--
A	C	15	$\tau_A(B) / 15$	$\tau_C(B) / \infty$
C	D	25	$\tau_C(A) / 10, \tau_C(B) / \infty$	$\tau_D(A) / \infty, \tau_D(B) / \infty$
A	D	45	$\tau_D(C) / 20, \tau_A(B) / 45$	$\tau_A(C) / 30, \tau_D(B) / 61$
B	D	50	$\tau_D(A) / 5, \tau_D(C) / 20$	$\tau_B(A) / 50, \tau_B(C) / \infty$
B	C	57	$\tau_B(D) / 7, \tau_B(A) / 20$	$\tau_C(D) / 32, \tau_C(A) / 65$

Table 6. Focus Phase section of Network Record Table with node D

Sender	Receiver	Time Stamp	Low Utility Value	High Utility Value
A	B	00	--	--
A	C	15	$\tau_A(B) / 15$	$\tau_C(B) / \infty$
C	D	25	$\tau_C(A) / 10, \tau_C(B) / \infty$	$\tau_D(A) / \infty, \tau_D(B) / \infty$
A	D	45	$\tau_D(C) / 20, \tau_A(B) / 45$	$\tau_A(C) / 30, \tau_D(B) / 61$
B	D	50	$\tau_D(A) / 5, \tau_D(C) / 20$	$\tau_B(A) / 50, \tau_B(C) / \infty$

If the node is malicious, it will try to come in path of the message delivery route by falsely declaring a lesser utility value. Now, let us assume that A and B encounter each other after 7 more time units, and B declares its utility value for D (where D is destination) as 5. Now, A and B follow the *focus detection algorithm*.

CASE-1: ***Detection of node B if it is malicious-*** From the point where contact of node B and node D is searched node A will check downwards in the table and check whether $\tau_B(D)$ has changed by transitivity or not. As we can see that it has not been changed, therefore at present its value should be 14 (7 when B and C meet after 7 sec, another 7 when we are assuming that A and B meet). Now the declared value is 5, calculated value is 14, hence B is malicious and A blacklists it.

CASE-2: ***Detection of C if it is malicious-*** D & C meet each other after 10secs of B & C meeting (at time 67). Let declared value of $\tau_C(A)$ be 26. D will search last encounter of A & C, then it moves downward and checks for any change in $\tau_C(A)$. It's evident that when B & C meet at 57, $\tau_C(A)$ changes from 65 to 28 and thus at 67, its value should be 38, which is greater than declared value; C is detected.

CASE-3: A case may arise where the regular node has a less updated NRT than a peer malicious node, which is not sufficient for detection. Even in that case, since nodes update NRTs at every encounter, it won't long before an encounter with a sufficiently updated regular node takes place, and the malicious node is detected.

Algorithm 2. Focus Phase Detection Algorithm

```
//Key: checker = checking node, peer = checked node
BEGIN
If checker & peer NRT entries match till last entry, then
              If peer has met destination before, then
                  If τpeer(destination) has changed by transitivity, then
                    Calculate actual value of τpeer(destination) using up-
                    date through transitivity values;
                  Else
                    Calculate actual value of τpeer(destination) using
                    time difference from last encounter;
                  EndIf
              EndIf
              If actual value of τpeer(destination) & declared value of
              τpeer(destination) do not match, then
                    Peer is malicious, append peer to MIL;
              Else
                    Follow Focus phase mechanism;
              EndIf
EndIf
END
```

4 Evaluation through Simulation

A. Simulation Setup & Methodology: We carried out relevant simulations using *ONE simulator*[5]; it was customized to impart malicious behaviour to randomly chosen nodes. Modifications were made to *Spray & Focus* code. Setup in Tab.7.

Table 7. Simulation Setup

Parameter	Value	Parameter	Value	Parameter	Value
Simulation Area	4500x3000	Packet Size	500KB – 1MB	Message TTL	480min
No. of nodes	100	Buffer Size	512 MB	No. of simulations	15/case
Mobility Model	Shortest Path Map Based	Interfaces (Range,Speed)	Bluetooth (10m,2Mbps), High-speed (1km,40Mbps)	Msg Generation Interval	25–35 secs
Simulation Time	12 hours	Node Speed	0.5m/s – 1.5m/s	Copies/msg	3

The following scenarios were considered: (1) Percentage of malicious nodes was varied among 10%, 20%, 30% & 40% of the total no. of nodes; (2) Total no. of nodes in the network was varied among 100, 90, 80, 70 & 60. In (1), we study the effect of varying percentage of malicious nodes on detection time & delivery prob. In (2), we perform comparative study between metrics obtained using (n,p) nodes & (n-p,0)

nodes where, (x,y) = (total no. of nodes, no. of malicious nodes). In each simulation, an appropriate no. of nodes have been chosen randomly and made to behave maliciously: 50% as *spray malicious*, rest as *focus malicious*.

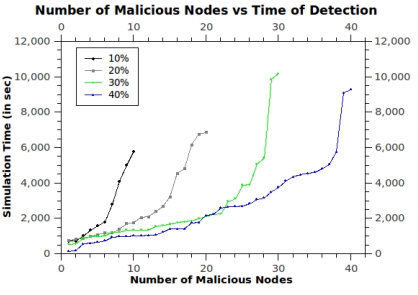


Fig. 1. Time v/s Average no. of detections

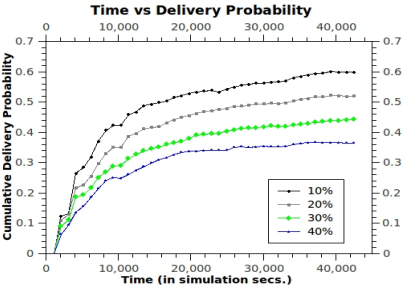
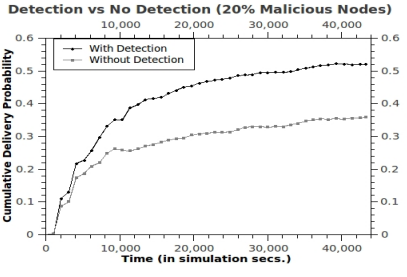


Fig. 2. Cumulative del. prob. v/s Time



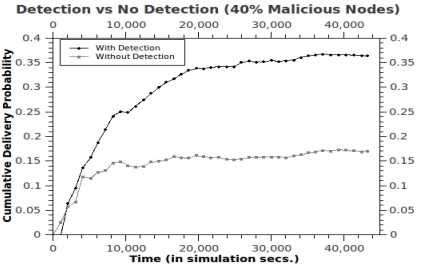
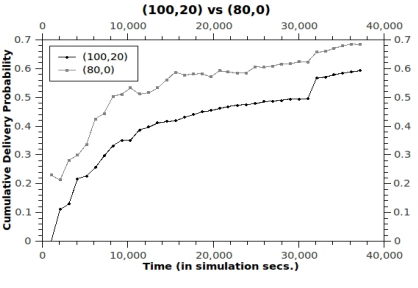


Fig. 3(a,b). Improvement in del. prob. in 20% & 40% cases using our proposed mechanism



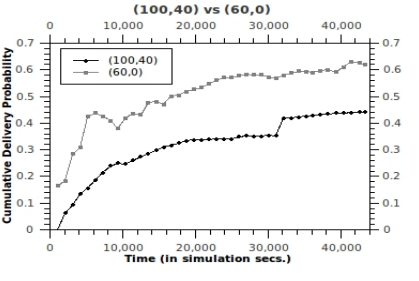


Fig. 4(a,b). Comparison between maliciousness inclusive & exclusive scenarios

B. Metrics Observed: The metrics used to evaluate performance and efficiency of the proposed detection strategy are: (1) *average number of detections* - it gives the avg. no. of detections made successfully at certain time intervals, thus showing how quickly our strategy is able to detect and eliminate malicious nodes; (2) *total delivery probability* - it gives ratio between total no. of messages delivered during entire simulation time & total no. of messages created during entire simulation time; (3)

cumulative delivery probability - it gives ratio between no. of messages delivered & no. of messages created till present simulation instance thus enabling us to study the pattern in which delivery ratio increases or decreases upon detection and elimination of more and more malicious nodes from the network.

C. Simulation Results and Analysis: The simulation results are as follows:

1. Efficiency of the proposed Detection Mechanism: Fig.1 presents the graph obtained by plotting *average number of detections* against time in seconds. We have plotted values obtained against all 4 cases of scenario (1). From the graph, we can conclude that the detection algorithm works effectively as it detects all malicious nodes in the network within satisfactory time. As expected, more the no. of malicious nodes in the network, more is the time required to detect and eliminate all of them. It is also noteworthy that detections occur very quickly when the simulation time is at lesser values, whereas as higher values of time,detection slows down. This can explained in the following way initially, node movement is less and the number of messages in the network is also quite meager; therefore it is easier to detect anomalies within NRTs due to lesser number of entries. On the contrary, after an appreciable period of time, as node mobility and message transmissions increase, it becomes more difficult to acquire sufficient knowledge about the entire network to detect anomalies.

2. Impact on delivery probability: Fig.2 depicts a comparative picture of delivery probabilities obtained in all 4 scenarios. Fig.3 consists of 2 different graphs, each representing the efficiency of our proposed strategy in improving delivery prob. The impact is most profound when malicious activity is rampant. If we look at any individual curve, we shall notice that it consists of 3 phases: phase-1 is when del. prob. is significantly low due to setback from malicious activities; phase-2 is when the network gradually recovers from the effect and shows improvement; phase-3 is when the nework is in a steady state and shows increase in del. prob. at a decreasing rate until it almost becomes parallel to horizontal axis.

3. Comparison between Maliciousness Inclusive & Exclusive Scenarios: Fig.4 deals with yet another absorbing aspect of performance of our proposed strategy. It depicts comparative curves between *(n,0)* & *(n-p,0)* scenarios. Such a study compares between situations when a network is simulated with *(n-p)* nodes all regular, against when a network is simulated with *n* nodes which effectively comes down to *(n-p)* after the *p* malicious nodes have been detected & eliminated. The curves show that our scheme achieves fairly close delivery probs. compared to normal scenario; difference increasing as density of malicious nodes increases.

4. Limitations: Having discussed the advantages of our detection scheme in terms of performance, let's take a look at the limitations and drawbacks involved. As can be observed from Fig.5, the overhead ratio incurred in case of detection using our detection mechanism is significantly higher compared to that incurred when the network runs with malicious activities but without detection. This is due to the extra computation time required during each encounter in the network.

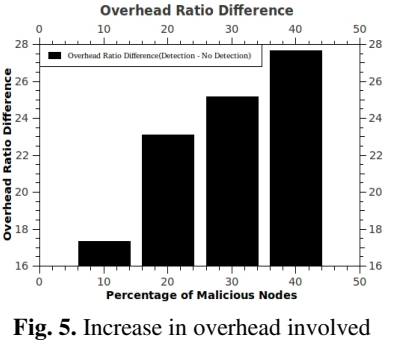


Fig. 5. Increase in overhead involved

5 Conclusion and Future Work

In this paper, we have taken a look at the proposed schemes which aim at combating various malicious attacks in a DTN environment. We have proposed a strategy for secured routing in the *Spray & Focus* routing scheme, which uses a table-based approach for detection of malicious nodes in the network. The intricacies of the mechanism have been described in detail. Simulations show us that the proposed scheme is effective in detecting malicious nodes and ensuring that those take no further part in the routing process. However, simulations also indicate significant increase in average latency and overhead involved. In order to curb such ill-effects, we are investigating a more balanced approach that allows only certain trusted checkers to perform *Maliciousness Tests*.

Other than the nature of attack considered and dealt with in this paper, there are a variety of attacks possible in a delay tolerant environment. Although a large amount of work has been carried out with regards to security in wireless ad-hoc networks, much remains to be done to combat similar and dissimilar security threats to disruption tolerant networks. Development of strategies to deal with a multitude of other attacks can be fodder for future research.

References

[1] Fall, K.: A Delay Tolerant Network Architecture for Challenged Internets. In: Proc. ACM SIGCOMM, pp. 27–34 (2003)
[2] Spyropoulos, T., et al.: Spray & Focus: Efficient Mobility-Assisted Routing For Heterogeneous & Correlated Mobility. In: Proc. Fifth IEEE PERCOM Workshops 2007 (2007)
[3] Lindgreny: Probabilistic Routing in Intermittently Connected Networks. In: Proc. ACM SIGMOBILE Mobile Computing & Communications Review, vol. 7 (July 2003)
[4] Burgess, J., et al.: Maxprop: Routing for vehicle-based disruption-tolerant networking. In: Proc. INFOCOM (April 2006)
[5] Kernen, et al.: The ONE Simulator for DTN Protocol Evaluation. In: Proc. of the 2nd Int'l Conf. on Simulation Tools & Techniques, Simutools 2009, Belgium (2009)
[6] Chuah, M., et al.: A Ferry based Intrusion Detection Scheme for Sparsely Connected Ad Hoc Networks. In: Proc. MOBIQUITOUS (August 2007)
[7] Ren, Y., et al.: MUTON: Detecting Malicious Nodes in Disruption-Tolerant Networks. In: Proc. IEEE WCNC (2010)
[8] Li, F., et al.: Thwarting Blackhole Attacks in Distruption-Tolerant Networks using Encounter Tickets. In: Proc. INFOCOM (2009)
[9] Ren, Y., et al.: Detecting Blackhole Attacks in Disruption-Tolerant Networks through Packet Exchange Recording. In: Proc. IEEE WoWMoM (2010)
[10] Zhu, H., et al.: SMART: A Secure Multilayer Credit-Based Incentive Scheme for Delay-Tolerant Networks. IEEE Trans. on Vehicular Tech. 58 (October 2009)

[11] Lu, R., et al.: Pi: A Practical Incentive Protocol for Delay Tolerant Networks. IEEE Trans. on Wireless Communications 9(4) (April 2010)
[12] Li, Q., et al.: Routing in Socially Selfish Delay Tolerant Networks. In: Proc. of INFOCOM, pp. 857–865 (2010)
[13] Ning, T., et al.: Incentive-Aware Data Dissemination in Delay-Tolerant Mobile Networks. In: Proc. of SECON (2011)
[14] Saha, S., et al.: Secured Routing in DTNs: Threats & Counter-measures. In: Ph.D. Forum, ICDCN (2011)