Performance Evaluation of Attack Detection Algorithms in Delay Tolerant Networks

Chaudhari Rajashri M.
PG Student
SSVPS's BS Deore College of Engineering,
Dhule, 424005,
India

Patil Manesh P.
Assistant Professor
SSVPS's BS Deore College of Engineering,
Dhule, 424005,
India

ABSTRACT
DTN (Delay Tolerant Network) is a new kind of wireless technologies which includes Radio Frequency (RF) and acoustic (sonar) technologies. DTN developed for interplanetary use where the speed of light is slow. DTN is a new kind of network derived from deep space communication. DTN is characterized as long delay and intermittent connectivity. The Delay Tolerant Network (DTN) is more vulnerable to different kinds of attacks like blackhole and greyhole attacks, due to limited connectivity. There is no end to end connectivity between source & destination in DTN. So that it uses store, carry and forward mechanism to transfer the data from one node to other node. Delay tolerant networks (DTNs) are characterized by delay and intermittent connectivity, due to this, malicious nodes drops all or a part of the received messages. This dropping behavior is known as blackhole and greyhole attacks respectively. Existing research scheme can detect individual attackers well but they cannot handle the case where attackers cooperate to avoid the detection. So that SDBG scheme implements an algorithm to detect individual attacks with collusion attack. The simulation result shows the protocol reduces the delivery delay using RAPID protocol by detecting collusion attacks that is simulated using the ONE simulator.

General Terms
Delay Tolerant Network, Blackhole attack, Greyhole attack, DTN routing.

Keywords
Delay Tolerant Network, Blackhole attack, Greyhole attack, Collusion, Detection Accuracy, and Delivery Delay.

1. INTRODUCTION
In Mobile Adhoc Network (MANET) packets can be transferred only if link between the nodes are established. If link is not established then packets will be lost. So packet delivery ratio will be decreased in MANET. To overcome this problem, Delay Tolerant Network (DTN) is used. In DTN each node has some storage capacity. So if the links of nodes are not established then packets will be stored in the storage. Communication services in unreachable & unfriendly environments are provides by DTN [11].

Delay-tolerant network addresses the issues regarding heterogeneous network that may lost network connection continuity. Examples are those networks operating in mobile or extreme terrestrial environments, or planned networks in space. DTN is a new kind of network derived from deep space communication. A series of contiguous networks data bundles that are defined by a new kind of network that enables applications. In DTN, there is no end to end connectivity between source and destination. DTN is characterized by long propagation delay and intermittent connectivity [12]. DTN is a set of protocols that acts together to enable a standardized method of performing store-carry-forward mechanism.

Individual and collusion attacks can be detected by Statistical-based Detection of Blackhole and Greyhole Attackers (SDBG) [1].

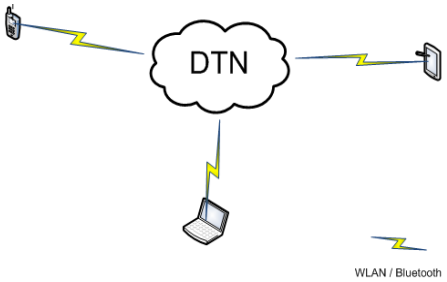

Fig 1: DTN Architecture

1.1 Challenges in Delay Tolerant Network
Specific challenges in an opportunistic network are: the contact opportunity and the node storage.

1. Node contact opportunity
A node might make contact with other nodes at an unpredicted time, due to the node mobility or the dynamics of wireless channel. For exchanging messages between some nodes that can move between remote fragments of the network must be exploited opportunistically and the contact between nodes is highly predictable. Two parameters, contact duration and inter-contact time that are important parameters in determining the capacity of an opportunistic network.

2. Storage constraint
To avoid dropping packets, the intermediate nodes are requires having enough storage to store all messages for an unpredictable period of time until next contact occurs. The required storage space increases a function of the number of messages in the network. The storage constraints are taken into consideration by routing and replication strategies. However, multiple-copy scheme generally incurs significant overhead on storage constraint.

3. No end-to-end path exists between source and destination all times.
4. MANET's routing protocols fail.
5. There is no knowledge about topology.

1.2 Security Requirements for DTN
1. Authentication
For every intermediate DTN node, it is essential to have the ability to check that the data sent by an authorized node. The data was sent at a legitimate rate and also asking for the class of service they are granted. The requirement of authentication

depends on goals of security design and provided either on a hop-by-hop or end-to-end basis.

2. Confidentiality
Confidentiality requirement is to ensure that sensitive information is not revealed to unauthorized third parties during the bundle propagation process over DTN links.

3. Integrity
Integrity requirement should ensure that the transmitted messages cannot be altered during the propagation process. Lack of integrity protection could result in many attacks including message modification, falsification, or replay attacks.

4. Privacy/ Anonymity
The network should not reveal the location of the user, nor the party with which she communicates.

2. RELATED WORK

In 2007, M. Chuah, P. Yang, J. Han was introduced FBIDM which is used with custody transfer feature when a multihop routing scheme used. This scheme doesn't perform well when history based routing schemes are used. So FBIDM can run history based routing schemes e.g. Prophet, Maxprop. The geographical area is divided into multiple cells and has ferries visit the center of each cell using some fixed routes, for the single ferry and two ferries. Each ferry stops at a few locations within its route. At each location, the ferry will broadcast a secret service message that each legitimate node knows deciphering [2].

In 2009, F. Li, J. Wu, and A. Srinivasan was developed a scheme, which is based on authenticated encounter records make it impossible for the adversary to claim non-existent encounters and abuse them to forge routing metrics to attract data. They defines Encounter tickets that are introduced for routing and packet forwarding, when two nodes meet, they generate an encounter ticket that carries a timestamp. Based on trusted PKI, two nodes sign the ticket with their private keys. When a node history reveals with another node, it submits the encounter tickets instead of a compressed list containing only node ID's and the number of contacts previously employed. This is the ticket based history interpretation scheme [3].

In 2010, Yanzhi Ren, Mooi Choo Chuah, Jie Yang, Ying ying Chen was developed Mutual correlation detection scheme (MUTON) to address insider attacks. The transitive property of MUTON considers when calculating the packet delivery probability of each node and correlates the information collected from other nodes. Each node collects the packet delivery probabilities of any node that it encounters with and past encounter history of that node. The collected information is used for estimating the changes in the delivery probabilities to other nodes. During detection, when the ferry encounters a node, it uses a self-examination approach [4].

In 2012, Qinghua Li and Guohong Cao proposed scheme that detects packet dropping in a distributed manner. A node is required to keep previous signed contact records i.e. the buffered packets that are sent or received and report them to the next contact node. This node can detect whether the node has dropped packets based on the reported records. To detect consistency, a small part of each contact record is disseminated to selected nodes, collect appropriate contact records and detect misbehaving nodes with certain probability [5].

In 2013, Y. Guo, S. Schildt, and L. Wolf were developed a scheme. This scheme is based on encounter records to estimate the forwarding ratios. It detects both blackhole and greyhole behaviors with high detection rate [6].

In 2013, N. Li and S. K. Das developed a scheme uses a distributed trust-based framework in which the forwarding behavior of a node is acknowledged by its next hop and a forwarding receipt is sent out to other nodes to update its reputation [7].

In 2014, Z. Gao, H. Zhu, S. Du, C. Xiao and R. Lu developed a probabilistic misbehavior detection scheme (PMDS) which is used to detect misbehavior in DTN, and collects relevant secured evidences like contact over network by investigating suspected node [8].

In 2016, Mythili M., Renuka K. Developed a scheme that detects different types of attack on DTN such as blackhole and greyhole attacks using fuzzy rule. This detection system is based on Fuzzy Logic. An IDS system is improved by making use of two factors i.e. packet loss rate, data rate. They use both factors with fuzzy logic to solve problem using problem solving control system. In this, a fuzzy algorithm is used to detect attack [9].

3. SYSTEM ARCHITECTURE

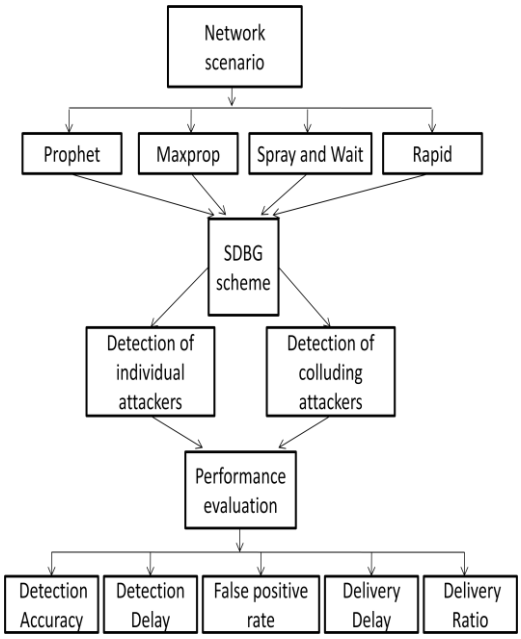

Fig 2: System Architecture

3.1 SDBG Scheme

Individual and collusion attacks can be detected by Statistical-based Detection of Blackhole and Greyhole attackers (SDBG). Nodes are required to exchange their encounter record histories, based on which other nodes can evaluate their forwarding behaviors. In an individual detection scheme, nodes are evaluated by their histories of encounters with other nodes. This history information is called as Encounter Record (ER). Forwarding Ratio (FR) is the ratio between total number of sent and received messages. A node is judged as malicious if its forwarding ratio is lower than the threshold.

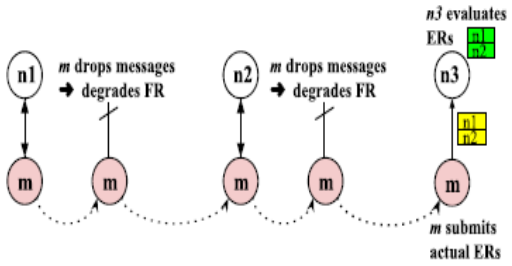

Fig 3: Individual attack

As shown in Figure 3 [1], n1 sends message to m, m drops that message so degrades FR. When n2 sends message to m, it drops that message so degrades FR. n3 can detect m as malicious when judging its authentic records including ER with n1, n2.

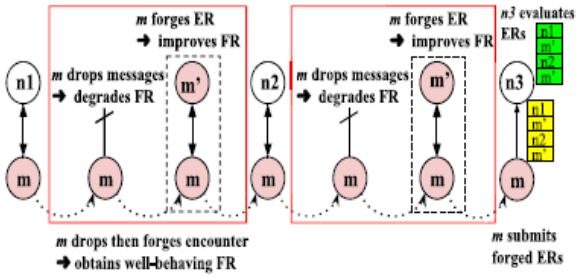

Fig 4: Collusion attack with individual detection scheme

As shown in Figure 4 [1], m can produce fake records to trick n3 to judge it as normal by colluding with m'. Attackers send out their own messages rather than messages of other nodes, to improve FR, to behave like a normal node. In collusion attack, attackers have to create fake encounter records to increase the forwarding ratio metric, so that their total number of sent messages can be high. This can be used to detect collusion attack.

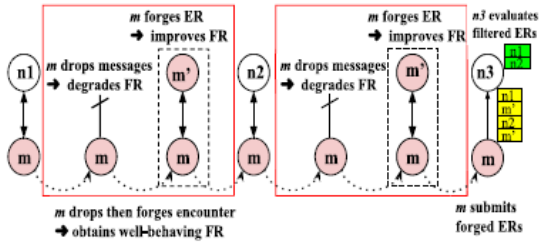

Fig 5: Collusion attack with SDBG

To avoid detection, m and its colluder m' have to create a large number of fake records. To reduce for dropping messages received from n1 and n2, these fake records have to include high number of sent messages. When it encounters n3, m submits the forged record window which includes contacts with n1, m', n2 and m'. For confirmation, n3 can exclude the suspiciously fake records between m and m' from the submitted history of m. The real forwarding ratios inferred from the authentic window allow n3 to detect m as malicious. This is shown in Figure 5 [1].

3.2 Detection of Individual Attackers

When dropping messages, malicious nodes only relay parts of the messages they receive and prefer sending messages for themselves to sending for other nodes. Each node maintains a local black list which lists malicious nodes that it detects as blackhole or greyhole attackers.

3.2.1 Manipulation of Encounter Records

According to the rule of creating ER, a series of consecutive well-behaved ERs has sequential sequence numbers. Besides, ER with higher sequence number has a bigger timestamp. When each node *a*, receives and processes an ER history of a neighbor *b*, it records the sequence number assignment by other nodes, not only by *b* but also by those encountering *b*. Once detected, the malicious node committing the inconsistency is blacklisted. The process does not incur additional communication overhead and only requires a small storage of sequence number history.

3.2.2 Dropping Misbehavior

Attackers might receive a lot of messages but only relay a small portion of them as the rest have been dropped intentionally. Selfish nodes send the messages, among that the large portion of messages are generated by them only and rest are the messages of other nodes. Dropping misbehavior can be notified with the Relaying Ratio (RR) and Self-forwarding Ratio (SFR) as follows.

$$RR1 = \frac{N1_{RS}}{N1_{RNS}} \quad (1)$$

$$SFR1 = \frac{N1_{self\_send}}{N1_{send}} \quad (2)$$

Where $N1_{RS}$ is the total number of messages, that *b* received and already relayed by forwarding to another node; $N1_{RNS}$ is the total number of messages that *b* received as a relay (not the ultimate destination) but b has not sent out. $N1_{self\_send}$ is the total number of messages that are generated and sent out by node *b*; $N1_{send}$ is the total number of messages sent out (regardless of the message source).

Basically, normal nodes have high RR and low SFR. If *RR1* falls below the threshold $Th1_{RR}$, the reputation of node *b* judged by node *a* will be decreased. If *SFR1* exceeds the threshold $Th1_{SFR}$, the reputation of node *b* is further reduced. However, if both thresholds are not violated, node *b* will have its reputation increased.

3.3 Detection of Collusion Attackers

Number of malicious nodes cooperates with each other to cheat the defense system, which misbehaves with the normal nodes and performs blackhole and greyhole attacks.

3.3.1 Collusion Packet-dropping Attack Model

The strategy for an adversary is creating fake encounter records with its colluders to manipulate its own metrics *RR* and *SFR*. Colluding attackers are assumed to know the private keys of one another.

$$RR1 = \frac{N1_{RS}}{N1_{RNS}} < Th1_{RR} \quad (3)$$

$$SFR1 = \frac{N1_{self\_send}}{N1_{send}} > Th1_{SFR} \quad (4)$$

If violating these threshold metrics, then the malicious message-dropping attackers can be identified. To avoid punishment, *m* needs to forge an ER so that the dropped portion is compensated and the new metrics over all its manipulated ER history are out of the abnormal range defined by the thresholds.

To hide violating thresholds, malicious nodes choose the parameters as follows, to behave like a normal nodes, by creating fake RR1' and SFR1'.

$$RR1' = \frac{N1_{RS+n1_{RS}}}{N1_{RNS}+ n1_{RNS}} \geq Th1_{RR} \quad (5)$$

$$SFR1' = \frac{N1_{self\_send+n1_{self\_send}}}{N1_{send+ n1_{send}}} \leq Th1_{SFR} \quad (6)$$

The newly created fake encounter record's parameters are denoted as: number of messages sent for itself $N1_{self\_send}$, number of sent messages $n1_{send}$, and number of received messages $n1_{recv}$. Suppose this fake ER makes the total number of messages received but not relayed increase by $n1_{RNS}$ and the numbers of messages received and relayed increase by $n1_{RS}$. The fake metrics *RR1'* and *SFR1'* are formulated as:

$$RR1' = \frac{N1_{RS+n1_{RS}}}{N1_{RNS}} \geq Th1_{RR} \quad (7)$$

$$SFR1' = \frac{N1_{self\_send}}{N1_{send+ n1_{send}}} \leq Th1_{SFR} \quad (8)$$

This is equivalent to:

$$n1_{RS} \geq Th1_{RR} \times N1_{RNS} - N1_{RS} \quad (9)$$

$$n1_{send} \geq N1_{self\_send} / Th1_{SFR} - N1_{send} \quad (10)$$

As $n1_{send} \geq n1_{RS}$ so the lower bound for $n1_{send}$

$n1_{send} \geq$ max
$\left(Th1_{RR} \times N1_{RNS} - N1_{RS}, N1_{self_{send}} / Th1_{SFR} - N1_{send}\right)$

If *m* sets $n1_{self\_send}$ and $n1_{recv}$ larger than 0, $n1_{send}$ even has to be larger.

3.3.2 Analysis of Collusion Behavior

A malicious node can manipulate its forwarding ratio by creating one single fake ER with number of sent messages $n1_{send}$ and the risk is that $n1_{send}$ might be suspiciously high. The attacker can create more than one fake records i.e. *l* fake ERs with the corresponding number of sent messages $n_1, \ldots, n_l$.

$$n1_{send} = n_1 + \ldots + n_l = \sum_{i=1}^{l} n_i \quad (11)$$

For each node *c* recorded as encountering *b* in ERs, denote $freq1_b^c$ as the number of encounters between *b* and *c* appearing in ERs, $send1_b^c$ as the total number of messages sent from *b* to *c* over the window. The metric *FXS1* is defined as

$$FXS1_b^c = freq1_b^c \times send1_b^c \quad (12)$$

If *b* and *c* are colluders, $FXS1_b^c$ reflects the abnormality of fake encounters between them; thus $FXS1_b^c$ should be abnormally high.

3.3.3 Detection of Collusion

Collusion message dropping attack is launched in a 2-phase process: suspicion and confirmation. . The first phase produces a list of suspicious colluders based on the *FXS1* metric. The second phase is required to avoid false accusation of the suspected node that has high FXS1 metric because it actually has many encounters and exchanges many messages with node *b*.

Following algorithm shows how the collusion detection is integrated with the individual attack detection.

Algorithm: Integration of Collusion attack detection with Individual attack detection

1. For Individual attack detection
   i. Calculating metrics Relaying ratio (RR1), Self Forwarding Ratio (SFR1) from Encounter Record Window (ERW).
   ii. Check if RR1 < threshold of RR1
   Then trust reputation TR is decreased by γ
   Set dropping = true
   iii. Check if SFR1 > threshold of SFR1
   Then TR is decreased by ρ
   Set dropping = true
2. For Collusion attack detection
   i. Generate list of colluders based on FXS1
   ii. If FXS1 > threshold of FXS1 then calculate RR1' and SFR1' metrics from ERW'
      1. If RR1' < threshold of RR1
      Then TR is decreased by γ
      Set collusion = true
      2. If SFR1' > threshold of SFR1
      Then TR is decreased by ρ
      Set collusion= true
3. If dropping =false and collusion=false
   Then TR is increased by λ

3.4 Contribution

The contribution to the project is to minimize delivery delay. Delivery delay is the time taken for a message to be transmitted over a network from source to reach the destination. To minimize the delay, RAPID routing protocol is used. RAPID protocol is based on the concept of utility function. The protocol attempts to replicate the packet whose replication reduces the delay by the most among all packets in its buffer.

RAPID (Resource Allocation Protocol for Intentional DTN) protocol

Rapid protocol has three core components: a selection algorithm, an inference algorithm, and a control channel. To determine which packets to replicate at a transfer opportunity given their utilities, a selection algorithm is used. To estimate the utility of a packet, the inference algorithm is used based on given the routing metric. The control channel propagates the necessary metadata required by the inference algorithm.

Protocol rapid(X, Y):

1. **Initialization**: Obtain data from Y about packets in its buffer.
2. **Direct delivery**: Delivery of the packets that are forwarded to the destination Y in decreasing order of their utility.
3. **Replication**: For each packet i in node X's buffer
   a. If i is already in Y's buffer (as determined from the data of Y), ignore i.
   b. Estimate marginal utility, δUi, of replicating i to Y .
   c. Replicate packets in decreasing order of $\frac{\delta Ui}{Si}$.

   Where δUi : Increase in *Ui* by replicating i
   Si : Size of i
4. **Termination**: End transfer when out of radio range or all packets replicated.

Utility function is used to minimize the delay.

$$Ui = \begin{cases} -D(i), D(i) \geq D(j)\ \forall j \in S \\ 0, Otherwise \end{cases} \quad (13)$$

4. PERFORMANCE EVALUATION

4.1 Simulation Using ONE Simulator

For evaluation of DTN routing and application protocols, the Opportunistic Networking Environment (ONE) simulator is designed. The programming language used in ONE simulator is Java. Different synthetic movement models and real-world traces are used to create scenarios and for implementing routing and application protocols, a framework is offered. In a single framework, a broad set of DTN protocol simulation capabilities are offered by a Java based ONE simulator, which is designed based on analyzing numerous DTN routing and application protocols. Modeling of node movement, routing and message handling and inter-node contacts are the main functions of the ONE simulator. Result collection and analysis are done through visualization, reports and post-processing tools. A visualization of the simulation state showing the locations, active contacts and messages carried by the nodes are displayed by the graphical user interface (GUI). To model the behavior of store carry-forward networking is the main focus of simulator [10].

4.2 Experimental Setup

Table 1 shows parameter settings for simulation. The number of nodes varies such as 40, 50 and 60 nodes. There are 4 routing protocols are used such as Prophet, Maxprop, Spray and Wait, and Rapid. Number of attackers is varying.

Table 1: Simulation Parameter Settings

Parameter	Values
Transmission range	10 meters
Travel area	450m×340m
Travel speed	10-50 kmph
Simulation time	43200 seconds (12 hours)
Network Interface	Bluetooth
Default Movement Model	Shortest Path Based
Message generation rate	25-35 seconds
Message size	500kB-1MB
Seed	5 times

Evaluation Parameters

1. Detection Accuracy: Percentage of malicious nodes that can be detected by normal nodes.

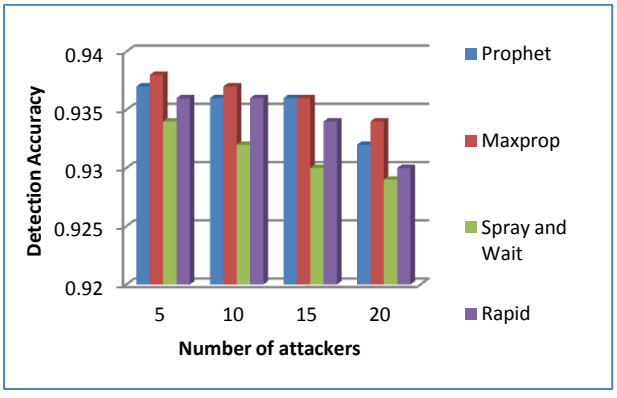

Fig 6: Detection accuracy for 40 nodes with Blackhole attack

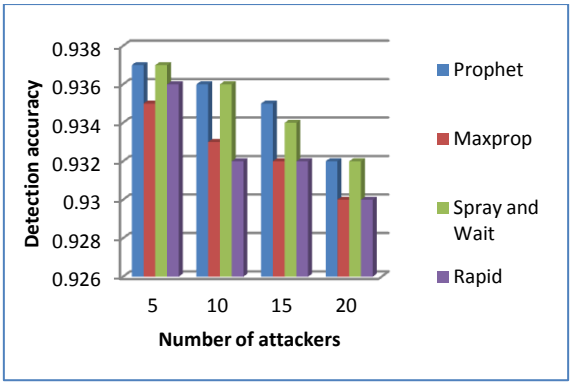

Fig 7: Detection accuracy for 40 nodes with Greyhole attack

2. Detection Time (Delay): The time taken for the misbehavior to be detected.

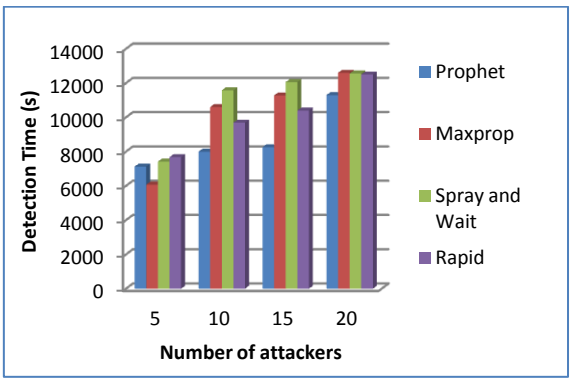

Fig 8: Detection time for 40 nodes with Blackhole attack

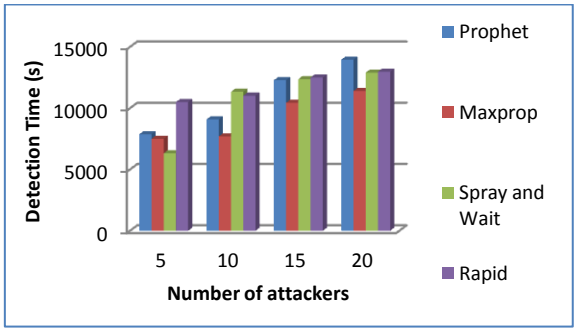

Fig 9: Detection time for 40 nodes with Greyhole attack

3. Detection False Positive Rate: Percentage of normal nodes that are mistakenly judged as malicious by other normal nodes.

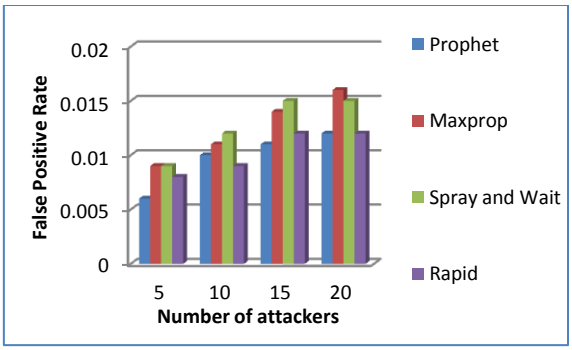

Fig 10: False Positive Rate for 40 nodes with Blackhole attack

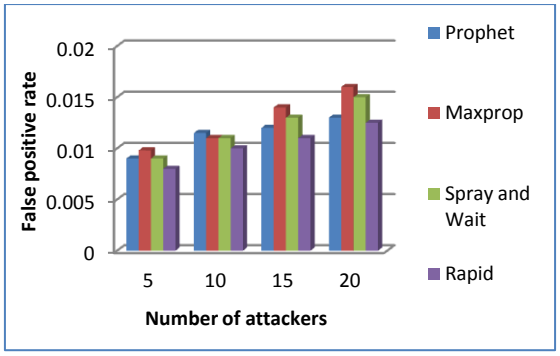

Fig 11: False Positive Rate for 40 nodes with Greyhole attack

4. Number of wasted transmissions: number of messages that malicious nodes have received from normal nodes and then dropped intentionally.

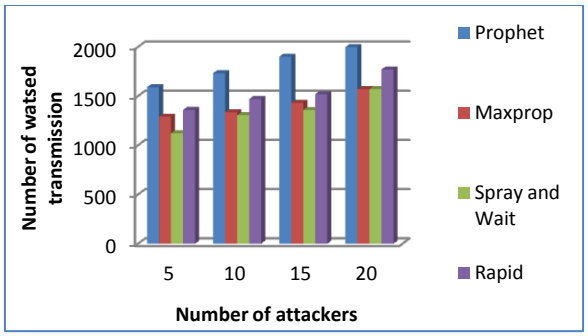

Fig 12: Number of wasted transmission for 40 nodes with Blackhole attack

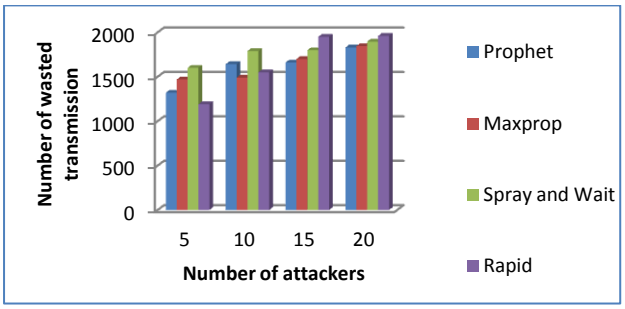

Fig 13: Number of wasted transmission for 40 nodes with Greyhole attack

5. Delivery ratio: percentage of messages delivered to destinations out of total generated messages.

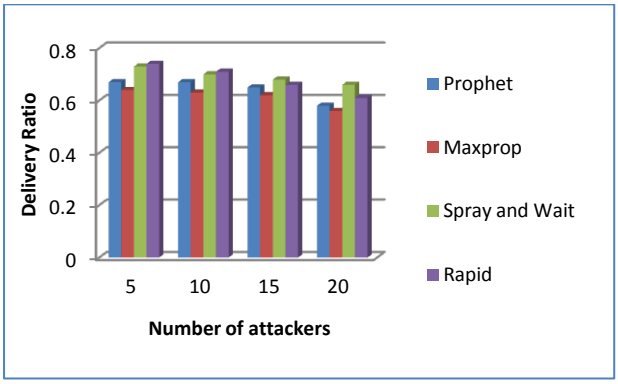

Fig 14: Delivery Ratio for 40 nodes with Blackhole attack

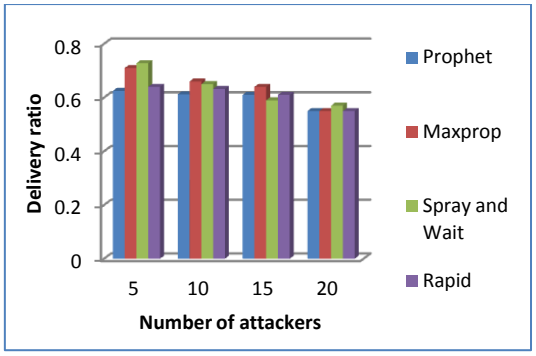

Fig 15: Delivery Ratio for 40 nodes with Greyhole attack

6. Delivery delay: average time taken for a message to reach the destination.

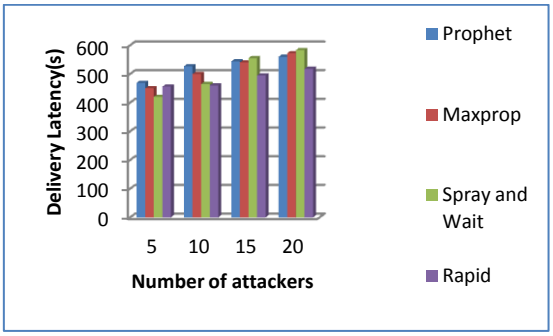

Fig 16: Delivery Delay (s) for 40 nodes with Blackhole attack

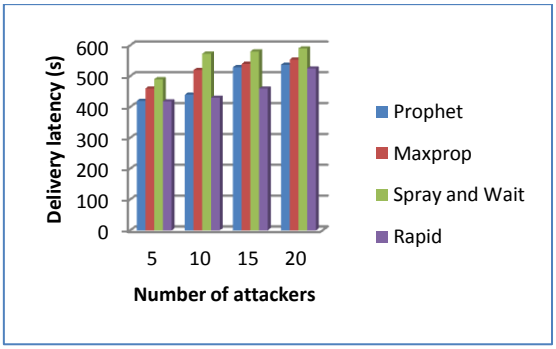

Fig 17: Delivery Delay (s) for 40 nodes with Greyhole attack

The following table shows, delivery delay minimization using Rapid protocol.

Table 2: Delivery Delay(s)

Routing Protocol / Attack type	Prophet	Rapid
Blackhole	497.5	495
Greyhole	513	510

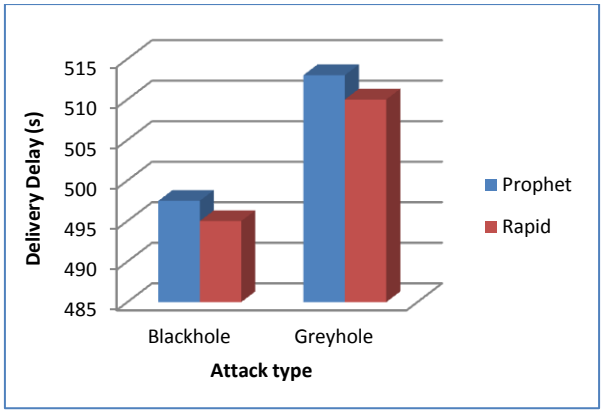

Fig 18: Delivery Delay(s)

5. CONCLUSION

Due to the limited connectivity, DTN is more vulnerable to the blackhole and greyhole attacks. The SDBG scheme, effectively defend against individual packet-dropping attacks in DTN. In SDBG scheme, the forwarding ratio is used to detect individual attack. This scheme is used to detect collusion attacks with high accuracy, high detection rate and low false positive rate. It improves message delivery rate by detecting blackhole and greyhole attacks effectively. The RAPID protocol is used to minimize delivery delay. The experimental result shows that delivery delay can be minimized by using RAPID protocol. In future, the scheme can be used for different kinds of network to detect different types of attacks.

6. REFERENCES

[1] Thi Ngoc Diep Pham and Chai Kiat Yeo, "Detecting Colluding Blackhole and Greyhole Attacks in Delay Tolerant Networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1116-1129, May 2016.

[2] M. Chuah, P. Yang, and J. Han, "A ferry-based intrusion detection scheme for sparsely connected ad hoc networks," in *In Proceeding 4th Annu. Int. Conf. Workshop Security Emerging Ubiquitous Computing*, 2007, pp. 1-8.

[3] F. Li, J. Wu, and A. Srinivasan, "Thwarting blackhole attacks in disrupt-tolerant networks using encounter tickets," *In Proceeding INFOCOMM*, pp. 2428–2436, 2009.

[4] Y. Ren, M. Chuah, J. Yang, and Y. Chen, "MUTON: Detecting malicious nodes in disrupt-tolerant networks," in *in Proceeding IEEE Wireless Communication Networking Conference*, 2010, pp. 1-6.

[5] Q. Li and G. Cao, "Mitigating routing misbehaviors in disruption tolerant networks," *IEEE Transaction on Information Forensics and Security*, vol. 7, no. 2, pp. 664-675, April 2012.

[6] Y. Guo, S. Schildt, and L. Wolf, "Detecting blackhole and greyhole attacks in vehicular delay tolerant networks," in *In Proceeding IEEE 5th international conference on Communication System and Networking*, 2013, pp. 1-7.

[7] N. Li and S. K. Das, "A trust-based framework for data forwarding in opportunistic networks," *Elsevier J. Ad Hoc Networking*, vol. 14, pp. 1497–1509, 2013.

[8] Z. Gao, H. Zhu, S. Du, C. Xiao, and R. Lu, "PMDS: A probabilistic misbehavior detection scheme toward efficient trust establishment in Delay-tolerant networks," *IEEE Transaction on Parallel and Distributed System*, vol. 25, no. 1, pp. 22-32, Jan 2014.

[9] Mythili M. and Renuka K., "An Efficient Black Hole and Gray Hole Detection Using Fuzzy Probabilistic Detection Scheme in DTN," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, no. 10, pp. 123-127, october 2016.

[10] A. Keranen, J. Ott, and T. Karkkainen, "The one simulator for dtn protocol evaluation," in *Proc. 2nd Int. Conf. Simul. Tools Tech.*, Rome, Italy, March 2009.

[11] (2014, february) ijareeie. [Online]. https://www.ijareeie.com/upload/2014/february/13l.html

[12] Thi Ngoc Diep Pham and and Chai Kiat Yeo. (2016, May) Detecting Colluding Blackhole and Greyhole attacks in Delay Tolerant Networks. ACM Digital Library.