# Monitor Potential Attack Locations in a Specific Area within DTN Network

IYAS ALODAT[1]

[1] Computer Network, Jerash University, Jerash, Jordan

Correspondence: IYAS ALODAT, Computer Network, Jerash University, Jerash, Jordan.

Received: February 7, 2021 Accepted: February 24, 2021 Online Published: March 16, 2021

doi:10.5539/cis.v14n2p42 URL: https://doi.org/10.5539/cis.v14n2p42

**Abstract**

In this paper will discuss and examine message transmission from the attacker process within the scope of Delay Tolerance Networks (DTNs). DTNs are a new area of research that can be developed in networking. Delay-tolerant networks are those networks that may not have a complete path between networks end-to-end via direct links and may be under development for a long time. As part of the improvement, we will compare a survey of DTN routing protocols with a real region area, and then taking into account the possibilities of detecting the presence of areas of weakness that lead to penetration, which will occur in the nodes while on the move. In this study, we will use the ONE simulator to track messages within nodes

**Keywords:** cyber security, DTN network, routing protocol, attacks

## 1. Introduction

Personal devices have a communication by voice and data when using mobile phone or laptop in ad-hoc networks, since the mobile devices are virtually always turned on and have radio interfaces, processor, data storage and battery usage. These points will make local connectivity among devices. However, in this case ad-hoc networks usually cannot support this kind of connectivity by using TCP/IP connection because of movement of the node in the area repeatedly; also networks divide when nodes move. Instead, We suggest passing messages by asynchronous through existing tracks in this kind of networks. For example, by using Delay-tolerant Networking (DTN) (Fall 2003, August), the performance of network will be different depending on how will node move, how dense the node population is, and how far apart the sender and the receiver are. Delivery time can change from a few minutes to hours or days, and a large portion of messages may not be delivered at all. The main factors are the routing and forwarding algorithms used, as well as the compatibility of design assumptions with actual mobility patterns.

WANs can be connected to each other through wireless or wired networks. The cell phone may connect to millions of servers scattered across the world (Ling 2007). But, all of these networks may not be able to reach everywhere, and may be expensive for some applications. These limitations may be the reason that current network technology relies on a set of basic assumptions that may be essentially legitimate. Some of the most important assumptions are that end-to-end communication exists between the source and the destination, possibly through multiple intermediaries due to the ability to provide power and unreliable networks. DTN are wireless networks where disconnection can occur due to frequent propagation delays, such as node mobility, power outages and more (Juang 2002 October). One of the main reasons for frequent outages is that the intermediate node randomly moves from point to point, the same node that carries data from source to destination. Delay and disruption-tolerant networks (DTNs) have characteristics of low connectivity; this is because it doesn?t have direct path connection from source to destination. In this kind of environment of routing protocols that depend on AOVD (Perkins & Royer 1999) and DSR (Johnson & Maltz 1996), it will fail to route messages. The main reason of failing message routing is that these routing protocols firstly find full path between source and destination before routing their data. Routing protocols must take store-and-forward to send their data, where data move and store through networks in hopes that eventually reach to destination (Burgess 2006), (Juang 2002 October), (Chaintreau 2007). A common technique that is used for successful data receiving is making replica of messages for hope to receive one of these replica messages (Vahdat 2000).

DTN is a model for computer networks and a rule system for transmitting information, often referred to as a set of protocols, which extends the capabilities of the Internet to the challenging communication environments of a space where the traditional Internet does not function well. These environments are typically subject to frequent disruptions, restricted one-way connections, and possibly long delays and high error rates.

DTN protocol suite can work together with IP group or it can operate independently. DTN provides guaranteed data delivery with automatic storage and forwarding mechanisms. Each data packet received is sent immediately if possible,

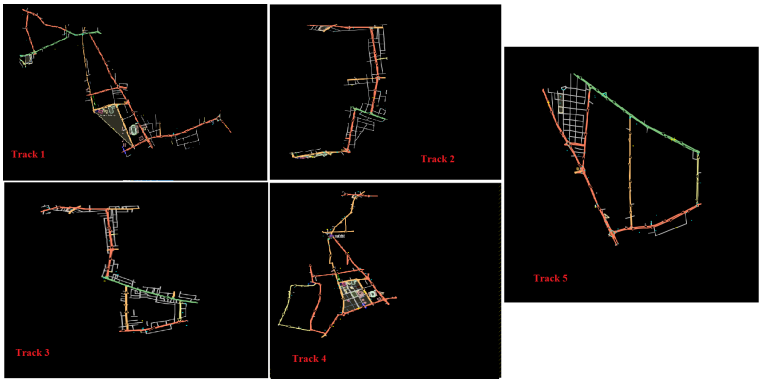


Figure 1. Roads divide with track of buses

but stored for future transmission if forwarding is not currently possible, but then expected to be possible in the future.

In the next sections, we will explain the types of attacks in section two, which will address three types of network attacks. And we will explain the relationship between deep learning and stenography in section three. In section four Simulation Setup, will handle the parameter used to build our simulation, and the environment. Finally, the results of the analysis and conclusion will be explained in sections five and six.

*1.1 Routing Protocol Classification*

Routing protocol taxonomy is based on replicas of messages, where routing protocol is not able to replicate messages that are considered as forwarding-base. While on the other side, replica is considered as replication-base.

Forwarding-base are less losing network resources, which is a result of return from the characteristic of sending message with no replica of that message, that means no other node will have this message (Jain 2004, August) (Henriksson 2007, August). These techniques will make most routing algorithm, but not all provide feedback from destination. However, we have such disadvantage of using forwarding-base that will appear from (Spyropoulos 2005, August) sufficient message delivery rate, which is not be allowed in many DTN. In the opposite direction, the replication-base allows much more message delivery rate (Burgess 2006). Most of DTN routing depends on heuristic-base making it less optimality.

1.1.1 Replication-base Routing

Although replica routing make better progress with message delivery ratios. It is possible with replication-based routing to include: network congestion in clustering areas, being wasteful with network resources (including bandwidth, storage and energy).

1.1.2 Epidemic Routing

Epidemic routing protocol (Vahdat 2000) contains flooding base techniques, when a node is joined or gets discovered in the network, the nodes will flood the packet data to these new nodes. Such kind of techniques may be used to limit the number of transferring messages.

1.1.3 PRoPHET Routing

In 2003 this strategy was documented (Lindgren 2003), using an algorithm that attempts to exploit non-random meeting in real world by keeping some probabilities of successful delivery to a known destination, and replicating messages only when having more than one opportunity to delivering it.

1.1.4. Direct Delay Routing

When the resource allocations have a problem in DTN routing, RAPID (Balasubramanian 2007 , August) comes to solve these problem by affecting the single routing metric. At time of publication, the goal is to minimize one of the following three metrics: Average delay, Missed deadlines, and Maximum delay. The overall protocol is composed of three steps: 1) Direct Delivery: Packets destined for immediate neighbors are transmitted; 2) Replication: Packets are replicated based on marginal utility (the change is utility over the size of the packet); and 3) Termination: The protocol ends when contacts break or all packets have been replicated.

## 2. Types of Attacks

There are three types of attacks (Tan, 2013, November): Black hole attack, flooding attack and selfish attack.

Figure 2. Irbid city with full roads

*2.1 Black Hole Attack*

Malicious nodes use routing protocols to get to the target by having the shortest path. The aggressive node will announce itself, so it is considered a new path regardless of any other way without verifying it. The attacking node has the ability to respond to this attack to quickly access the path. When using a flood dependent protocol, the response will be received from any physical node, so it will be a malicious and bogus path.

*2.2 Flooding Attack*

Flooding is a denial of service (DoS) attack aimed at stopping large amounts of network traffic from flooding a network or service. Flood attacks happen when a network or service floods packets that initiate incomplete connection requests and then the original connection requests cannot be processed. By flooding the server or host with a connection that cannot be completed, the flood attack fills the host's memory buffer. If this buffer is completed, and the effect is denial of service, thus no more connections can be made.

*2.3 Selfish Attack*

A node known as the selfish node based on most of its neighbors reports will be excluded from the group, and all cluster nodes will be informed of a message from the block head. Detecting the selfish node in the network will increase the throughput of the network and reduce the power consumption of the nodes. As a result, these well-behaved nodes experience drastic throughput drop, a large amount of delay, and a packet delivery ratio decrease.

## 3. Steganography and Deep Learning

Steganography is the strategy of concealing confidential data in a normal and non-confidential file or message to escape confidentiality to arrive at destination (Sallee 2003, October). Using details in digital images is the most common approach. We all know that digital images, like a JPEG image, contain many megabytes of pixel-like data. This makes some space for anyone to include data about steganography in the digital file. The hacker changes the less important sections of the data file with steganography applications and embeds malicious code in the picture. The malware is enabled until the target user downloads the image file and opens it on his device. To detect malicious code in images, we use deep learning techniques to discover malware hidden inside images.

## 4. Simulation Setup

In essence, the ONE Simulator is an independent agent-based event simulation engine (Kernen 2009, March). At each step of the simulation, the engine updates a series of modules that perform key simulation functions. The functions of the ONE Simulator are modeling node movement, contacting between nodes, routing and message processing. The

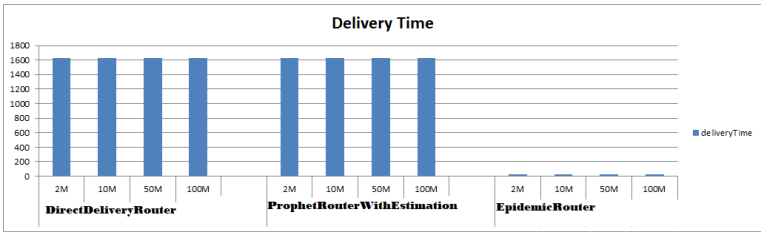


Figure 3. Estimation Ttl messages from source to destination in routing protocols

results are collected and analyzed through visualization, reports and post-processing tools. A detailed description of an emulator is available on the project page (Keranen 2008) and the ONE Simulator (Kernen 2009, March), the source code is also available. The movement of the node is done by movement models. This is either artificial models or current movement effects. Communication between nodes is based on their location, communication range and bit rate. The routing functionality is performed by routers that specify which messages are forwarded through existing contacts. Finally, the same messages are generated by the event generators. Unicast messages have a single host of origin and destination within the world of simulation. The results of the simulation are collected mainly through reports generated by report modules during the execution of the simulation. Unit Report Receive events (e.g., messages or call events) is edited from the simulation engine, which generates results accordingly. The results generated can be event logs that are then processed by external post-processing tools, or they can be aggregated statistics calculated in the simulator. Finally, the graphical user interface (GUI) shows a visualization of the simulation status that shows the locations, active contacts and messages that transport the nodes.

In Figure 1, the simulation was taken from map roads of Irbid city in Jordan (Wagner AGSE 2009). We fixed tracks of buses as in real world, as well as random tracks of personal cars going to work and shopping with some spot for stopping.

Moreover, we added smartphone cars, which were suggested for move as random with shortest path, as shown in Figure 2.

*4.1 Simulation Parameters*

In Figure 2, we use Well-known text (WKT) converting from eXtensible Markup Language (XML) languages (Wagner AGSE 2009). The WKT conversion of streets, detects and removes partitioned map parts and fixes any missing data.

We use WKT Tools (Mayer 2010) to divide our main city map with roads to determine them with our simulation. Then, we put our divided roads with ONE simulation for test routing under this city. The ONE simulator for testing DTNs in real world has an open source simulation that works under Java programming JDK. However, we run our simulation under Windows 10, Processor Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz, RAM 8.00GB. We simulate in area with three different routing protocols, as well as with different buffer sizes.Thus, the settings in simulations will change with routing Direct Delay Router, Prophet Routing, and Epidemics Routing. Moreover, buffer size will put 2M, 10M, 50M and 100M. Therefore, we will have 12 of result groups that we will compare between them. Number of hosts in the network area totally is 1055 distributed with 7 groups with characteristics, as shown in Table 1. From Table 1, we can see the last column has files that depend on track, which we pointed about before in Figure 2. In addition, all groups have the same characteristics that were not mentioned at Table 1, such as transmit range, transmit speed, wait time, routing protocol, buffer size and speed.

In Table 1, we can see group nr.6 and nr.7 with different kind of movement model, which have special characteristics like number of offices, work day length, probability stopping at shopping after work, office wait time coffee, office min wait time, office max wait time and number of meeting spots. These characteristics are being used by group nr.6. For group nr.7, we chose 1000 hosts that will work randomly as smart cars. These smart cars may be a special government car, a car using a smart application or taxi car,etc.

## 5. Analysis Results

From results of reports generated after each simulation, we observed and studied messages that go from host to host with each routing group we have chosen.

In Figures 3,4,5 , we can observe that Epidemic routing takes less delivered time. Moreover, Estimation Ttl for receiving messages, will give Epidemic routing a point advantage if we want to put speed as our priority in our network with sending and receiving messages. However, as we see the hop count takes more delivered time in Epidemic routing than Pprophet

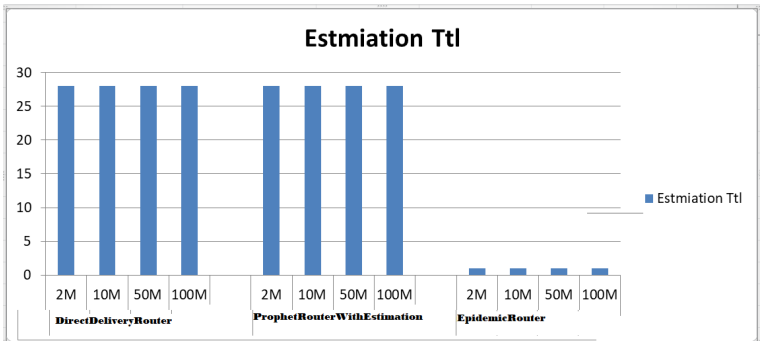


Figure 4. hop count messages in routing protocol

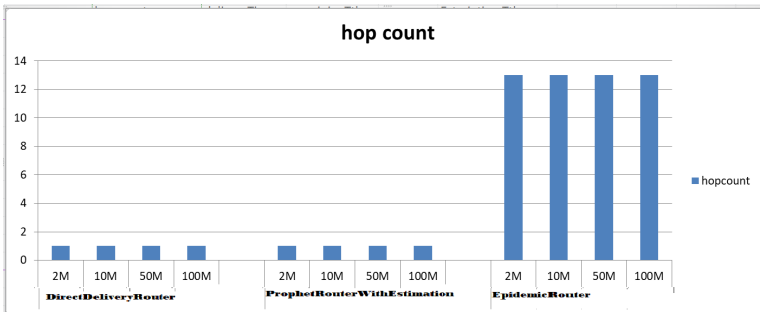


Figure 5. Delivered time messages in routing protocols

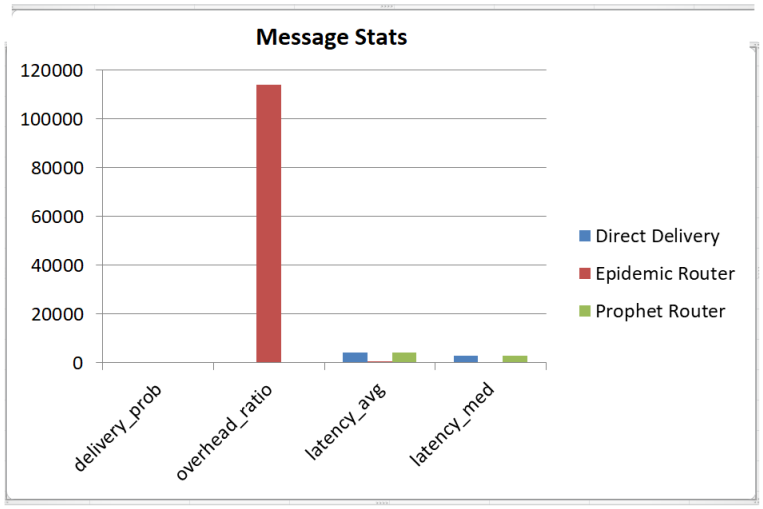


Figure 6. Message states in each routing.

Table 1. distribute node and characteristics

| Groups | | | |
|---|---|---|---|
| | 1,2,3,4,5 | 6 | 7 |
| Copy transmit range | 100 | 500 | 500 |
| Movement model | Bus Movement | Working Day Movement | Shortest Path Map Based Movement |
| Nr.of hosts | 1 | 50 | 1000 |
| Route file sequentially | - OldBusToNewBus.wkt<br>- NorthYu.wkt<br>- A3oarToNewBus.wkt<br>- NorthToHospitalBadiaa.wkt<br>- NorthToNewBus.wkt | NaN | NaN |

Table 2. Average message stats in each protocol routing

| | Delivery probability | Overhead ratio | Latency average | Latency med |
|---|---|---|---|---|
| Direct Delivery | 0.80015 | 0 | 4196.6766 | 2849 |
| Epidemic Router | 0.1615333 | 113925.424 | 538.95293 | 387.333 |
| Prophet Router | 0.80015 | 0 | 4196.6766 | 2849 |

routing and Direct Delivery routing. In other words, with epidemic routing our messages will be send and received to distension with much more than one hop. Therefore, this property will make the network treats with less transaction sensitive data, which means the network will be more risk. However, monitoring the network will manage this problem. As we see, if the network incurs with selfish attack, we will observe that by looking at deliver ratio and large amount of delay.

Direct delivery routing and Prophet routing take the advantages from Epidemic routing, as illustrated in Table 2. Overhead ratio is equal to zero, as shown in Figure 6. In both cases (Direct Delivery routing and Prophet routing), we can see the delivery probability is more than Epidemic routing. In other words, when the network has a lot of messages to send, the probability of Epidemic routing is less capable than the other routing to achieve this task. However, as we can see before in Figures 3,4,5 that Epidemic routing will send faster messages than others, which explains why latency average and latency med has an advantage in Epidemic routing protocol. However, at this notion, we monitor the network buffer to manage from attacker, as we see if the network incurs with flooding attack, we will observe that by looking at filling memory buffer.

In addition, buffer time was reserved until message arrives to destination. During scenarios, we observe the results as in Table 3. Epidemic routing algorithm will use buffer time more than Prophet routing protocol and direct deliver routing protocol. The reason confined using flooding base techniques for sending their data in the network, this is the main reason that makes epidemic routing protocol uses more buffer time in all kinds of buffer size, on the contrary of prophet routing protocol or direct deliver routing protocol that use buffer time only in the small buffer size 2M. However, in small buffer size Prophet routing protocol and Direct deliver routing protocol take long time, since the movement is directly to destination.

Table 3. Message status takes buffer time

| Buffer size | 2M | 10M | 50M | 100M |
|---|---|---|---|---|
| Direct Delivery | 18537.91 | NaN | NaN | NaN |
| Epidemic Router | 5.9936 | 29.26 | 146.3969 | 392.7938 |
| Prophet Router | 18537.91 | NaN | NaN | NaN |

## 6. Conclusion

After comparing between results, we can have such suggestion for using algorithm in our networks in the city. We can see that Epidemic routing protocols have more speed when looking at delivering messages, but delivery probability has the lowest one. When we look at time to live messages, we can see that epidemic routing protocol has a good result, at the same time it has much more overhead ratio compared to Direct delivery routing or Prophet routing protocols.

In Epidemic routing, the hop count messages in routing protocol has more than one hop to reach messages from source to destination; this is one of the reasons why time to live have best results.

From the analysis above, we use Prophet routing protocol or Direct delivery routing protocol even if they have more delivery probability and latency average, because since the network in these both protocols has more stability than Epidemic routing protocol, as well as overhead ratio is higher in Epidemic and in Prophet and equal to zero in Direct delivery routing protocol.

Upon of security issue to monitor our network against attacker, we can monitor large amount of delay in Prophet routing protocol and Direct delivery routing protocol, as well as delivery ratio to not be higher than usual. In black hole attack or flooding attack, we have to monitor buffer size if host will fill down from it.

## References

Balasubramanian, A., Levine, B., & Venkataramani, A. (2007, August). *DTN routing as a resource allocation problem. In Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications* (pp. 373-384).

Burgess, J., Gallagher, B., Jensen, D. D., & Levine, B. N. (2006, April). MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. *Infocom, 6.*

Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R., & Scott, J. (2007). Impact of human mobility on opportunistic forwarding algorithms. *IEEE Transactions on Mobile Computing, 6*(6), 606-620.

Fall, K. (2003, August). A delay-tolerant network architecture for challenged internets. In *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications* (pp. 27-34).

Henriksson, D., Abdelzaher, T. F., & Ganti, R. K. (2007, August). A caching-based approach to routing in delay-tolerant networks. In *2007 16th International Conference on Computer Communications and Networks* (pp. 69-74). IEEE.

Jain, S., Fall, K., & Patra, R. (2004, August). Routing in a delay tolerant network. In *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications* (pp. 145-158).

Johnson, D. B., & Maltz, D. A. (1996). Dynamic source routing in ad hoc wireless networks. In *Mobile computing* (pp. 153-181). Springer, Boston, MA.

Juang, P., Oki, H., Wang, Y., Martonosi, M., Peh, L. S., & Rubenstein, D. (2002, October). Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet. In *Proceedings of the 10th international conference on Architectural support for programming languages and operating system*s (pp. 96-107).

Keranen, A. (2008). *Opportunistic network environment simulator.* Special Assignment report, Helsinki University of Technology, Department of Communications and Networking.

Kernen, A., Ott, J., & Krkkinen, T. (2009, March). The ONE simulator for DTN protocol evaluation. In *Proceedings of the 2nd international conference on simulation tools and techniques* (pp. 1-10).

Lindgren, A., Doria, A., & Scheln, O. (2003). Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE mobile computing and communications review, 7*(3), 19-20.

Ling, C., Hwang, W., & Salvendy, G. (2007). A survey of what customers want in a cell phone design. *Behaviour & Information Technology, 26*(2), 149-163.

Mayer, C. P. (2010). *osm2wkt-OpenStreetMap to WKT Conversion. mayer2010osm, from OpenStreetMaps-ONE.*

Perkins, C. E., & Royer, E. M. (1999, February). Ad-hoc on-demand distance vector routing. In *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications* (pp. 90-100). IEEE.

Sallee, P. (2003, October). Model-based steganography. In *International workshop on digital watermarking* (pp. 154-167). Springer, Berlin, Heidelberg.

Spyropoulos, T., Psounis, K., & Raghavendra, C. S. (2005, August). Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant*

*networking* (pp. 252-259).

Tan, S., & Kim, K. (2013, November). Secure Route Discovery for preventing black hole attacks on AODV-based MANETs. In *2013 IEEE 10th International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing* (pp. 1159-1164). IEEE.

Vahdat, A., & Becker, D. (2000). *Epidemic routing for partially connected ad hoc networks.*

Wagner, D., Zlotnikova, R., & Behr, F. J. (2009). XML-BASED AND OTHER GEORELATED ENCODINGS: OVERVIEW OF MAIN EXISTING GEOCODING FORMATS. *AGSE, 196.*