



## International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

# Effective Transmission of Bundle in Delay Tolerant Networks

Leona Antony, Litty Antony

M.Tech Student [Cyber Security], Sree Narayana Gurukulam College of Engineering, Kerala, India

**ABSTRACT:**The delay-tolerant-network (DTN) model is becoming an effective communication than traditional model for mobile electronics connected with short-range wireless communication like Bluetooth, NFC, and Wi-Fi. Delay-tolerant networking (DTN) is a solution to possibly unpredictable loss of connectivity, long or variable delay, asymmetric data rates, and high error rates in heterogeneous networks. Security issues has been a major focus of the DTN. Due to the inappropriate security solutions, DTNs are often results in insecure data transfers. This paper proposes an effective malware detection scheme as well as bundle security for each bundle (packet) that is transferred across each node in Delay Tolerant Network

**KEYWORDS:** Bundle Security Protocol, Delay Tolerant Networks, Digital Signature, E Top-and-Tail Virus Scanning,

### I. INTRODUCTION

Delay-tolerant networking (DTN) is an approach to computer network that solves the technical issues in heterogeneous network that face the unpredictable loss of continuous network connectivity, long delay, high error rates. Examples of such networks are those operating in extreme terrestrial environments, or networks in space. Security issues has been a major focus of the DTN.

Due to the inappropriate security solutions, DTNs are often results in insecure data transfers. Virus attacks in such networks can lead to the issues such as confidentiality, integrity and availability in the data that are being transferred between nodes in the network. Denial of Service is a type of attack that can delay the availability of data across nodes. Another type is the Man-in-the-Middle attack in which an attacker can take over the data packet, by modifying or by capturing the packets which is being transferred between nodes in a network.

In 1987, Fred Cohen, the pioneer researcher in computer viruses, defined a computer virus to be: A program that can infect other programs by modifying them to include a possibly evolved copy of itself". Figure 1, is Dr.Cohens' pseudo-code of a simple computer virus. This is a typical structure of a computer virus which contains three subroutines. The first subroutine, infect executable, is responsible for finding available executable files and infecting them by copying its code into them. The subroutine do-damage, also known as the payload of the virus, is the code responsible for delivering the malicious part of the virus. The last subroutine, trigger-pulled checks if the desired conditions are met in order to deliver its payload.

Virus based on the DTN model brings unique security challenges that are not present in the traditional infrastructure model. In the infrastructure model, the cellular carrier in central watch the networks for any kind of malfunctioning. We can't assure about the data that is being passed between the nodes reached the other node or not securely. We here proposes an interdisciplinary mechanism such as Top-and-Tail virus scanning method as well as the Bundle Security Protocol in order to arrive at a strengthen Delay Tolerant Network.

There are simple methods to detect computer viruses [1]. Techniques usually involve scanning for pre-defined sequences of bytes called strings. Scanning only the first and the last 2, 4, or 8 KB of a file for each possible position is a good way to make virus detection much faster. This technique is called top-and-tail scanning and is used in this proposed scheme to optimize scanning speed by reducing the number of disk reads and well as to assure confidentiality, integrity and availability factors. Top-and-tail scanning became popular as the majority of early computer viruses prefixed, appended, or replaced host objects.

In efforts to provide a shared framework for algorithm and application development in DTNs [2], RFC 4838 and RFC 5050 were published in 2007 which defines a software running on disrupted or delayed networks, commonly known as the Bundle Security Protocol (BSP), this protocol defines a series of contiguous data blocks as a bundle where each bundle contains enough semantic information to allow the application to make progress where an individual



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

block may not. BSP provides data integrity and confidentiality services for the Bundle. Separate capabilities are provided to protect the bundle payload and additional data that may be included within the bundle. Bundle Security Protocol Specification includes:

1. Bundle-A bundle is a protocol data unit of the DTN bundle security protocol.
2. Bundle payload - A bundle payload (or simply "payload") is the application data of a file.
3. Fragment - A fragment is a bundle in which payload block contains fragments of payload.
4. Bundle node - A bundle node (simply a "node") is any end machine that can send and/or receive bundles.
5. Deletion, Discarding - A Bundle Security Protocol "discards" a bundle by simply ceasing all operations on the bundle and functionally erasing all references to it.

## II. RELATED WORK

Pirzada et al. [3] proposed a "Reputation System in MANETs" to protect the reactive routing protocol from attackers and increase the performance of the network. The review of these secure routing protocols for MANETs indicates that these protocols either use the Watchdog mechanism or ACK messages to build trust values between the nodes. In MANETs, a node can be evaluate another by using either direct or indirect measurements. The direct measurement is either achieved by using the watchdog mechanism or by using the ACK from the destination. Kejun Liu and Jing Deng [4] proposed "Reputation values constructed using the ACK messages sent by the destination node". These techniques are not applicable to DTNs due to the following reasons. In DTNs, a node cannot use the watchdog mechanism and monitor another intermediate node after forwarding its packets to it. This is because links on an end-to-end path do not exist contemporaneously, and hence an intermediate node needs to store, carry, and wait for opportunities to transfer those packets. As a result, the node loses connection with the intermediate node which it desires to monitor. K. Aberer and Z. Despotovic [5] proposed the "Work on the use of reputation systems for P2P networks". However, reputation systems for P2P networks are either not applicable for DTNs or they require excessive time to build the reputation values of the peers. Most proposed P2P reputation management mechanisms utilize the idea that a peer can monitor others and obtain direct observations or a peer can enquire about the reputation value of another peer (and hence, obtain indirect observations) before using the service provided by that peer. However, neither of these techniques is practical for DTNs. In DTNs, direct observations are not possible as we discussed above. Further, enquiring about the reputation value of a peer is not practical in DTNs due to opportunistic communications during contact times and intermittent connectivity of the peers. Vellambi and F. Fekri [6] proposed the "Challenges of providing secure communication in DTNs". Here the use of Identity-Based Cryptography (IBC) is suggested. Source authentication and anonymous communication as well as message confidentiality are provided using IBC. The use of packet replication is proposed to improve message delivery rate instead of using cryptographic techniques. They note that the existing techniques to secure DTNs are aimed to provide data confidentiality and authentication only. Jing Su et al. [7] suggested an "Effective countermeasure to a Bluetooth worm infection" which makes the device non discoverable or even turning the Bluetooth radio off. They find this solution unappealing: it will prevent devices from using Bluetooth for legitimate applications. Although Bluetooth worms spread orders of magnitude more slowly than Internet worms, Bluetooth worms spread quickly enough that human-mediated counter-response solutions are likely to be difficult to implement in practice. Based on their simulations, such solutions must detect a worm's presence, analyse infected code, and create, test, and distribute a security patch in the span of several days.

## III. PROPOSED SYSTEM

Though the previous papers written for securing the DTN was secured with IBC [9], there are of course some more security issues in a DTN. They are:

- Improper Key Management
- Distribution of private keys
- Distribution of system wide public parameters
- Replay attacks
- Lack of Virus Detection mechanism

### A. Key Generation using Elliptic Curve Cryptography

The proposed system eliminates those vulnerabilities by the mechanism described below: Need of proper Key management has been resolved using Elliptic Curve Cryptography. Elliptic Curve Cryptography (ECC) systems as

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

applied to cryptography were first proposed in 1985 independently by Neal Koblitz and Victor Miller. An elliptic curve is defined with: A finite field, usually consisting in integers modulo some prime  $p$  (there are also other fields which can be used). A curve equation, usually  $y^2 = x^3 + ax + b$ , where  $a$  and  $b$  are constant values from the finite field.

The curve is the set of pairs of values  $(x, y)$  which match the equation, along with a conventional extra element called "the point at infinity". Since elliptic curves initially come from a graphical representations (when the field consists in the real numbers  $R$ ), the curve elements are called "points" and the two values  $x$  and  $y$  are their "coordinates".

Suppose Alice wants to send to Bob an encrypted message. Alice and Bob creates public/private keys by agreeing on a base point,  $F$ . For that, Alice selects a secret 'a' and computes Public Key,  $PA = a * F$ . Similarly, Bob selects his own secret called 'b' and his Public Key,  $PB = b * F$ . They then exchanges their public keys to each other in order to create their corresponding private keys. When Alice gets the public key of Bob, she computes her private key,  $PrA = a * PB$ . Similarly, Bob computes private key,  $PrB = b * PA$  when he gets Alice's public key. Now, each nodes will generate their own public, private key pairs.

### B. Bundle:

The DTN Bundle Security Protocol (DTNBSP) defines the format and processing of the blocks used to implement the Bundle Protocol (Bundle), which is then simplified for this proposed system. The bundle (packet) consists of Primary Block and Payload Security Block (PSB) which is depicted in the Figure 2 and 3.

Type	Processing Flags	Cipher Suit
Source EID		
Destination_EID		
Life Time		
Length of Payload		
Payload(Fragmented)		
Digital Signature (Fragmented)		
Security result data		

Figure 2: Bundle Format

Bundle_proc_status	Bundle_life_status	Bundle_frag_status
--------------------	--------------------	--------------------

Figure 3: Processing flags

Fields from 'Type' to 'Length of Payload' is known as 'Primary Block' and the rest is known as 'Payload Security Block'(PSB). Below is the detailed description about each field in a Bundle.

### C. File Sending (At the sender side)

Suppose Client A and B joined in a Delay Tolerant Network. Now, Client A requests for a connection with the Client B if he wishes to send a file to B, Client B in response, checks for its availability and then accepts the request. In the meantime, both the clients agrees up on a public key and generates their own private keys using Elliptic Curve Cryptography (ECC) algorithm (as described earlier).

Now, the public key of Client A is available with Client B and vice versa. Client A then picks the file that he wishes to send. He then packs the file inside a Bundle using the Bundle Security Protocol (described earlier). The in depth explanation how data is packed inside a bundle will be explained in the next section. After selecting the file to be transferred, Client A starts forming the primary block and Payload Security Block of the bundle. Primary Block is explained below.

1. Type: Explains the type of protocol used to transfer a file in DTN. Normally, it will be a string value called "Bundle Security Protocol" and is added against this field.
2. Processing Flags: The field which decides whether the Bundle to be processed or not. It has 3 sub flags. They are:

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

- a. Bundle\_frag\_status: If the file is in a fragmented (divided into more than 2 parts) structure, this field is set to 1, else set to 0.
  - b. Bundle\_life\_status: initially it is set to 0. It changes only when it reaches the destination.
  - c. Bundle\_proc\_status: initially it is set to 0. It changes only when it reaches the destination.
  3. Source End Point ID (Source\_EID): This field contains IPv4 (32 bits) IP address of the source machine (here Client A's ip address).
  4. Destination End Point ID (Dest\_EID): This field contains IPv4 (32 bits) IP address of the destination machine (here Client B's ip address).
  5. Life Time: Client A adds the system's time in which the Bundle has been formed and some seconds needed for the file transfer and store them as byte.
  6. Length of Payload: Length of the file content (Payload) in bytes.
  7. Cipher Suit: contains name of algorithms used for encryption/decryption and hashing.
- Next, fields comes under Payload Security Block is explained below.
8. Payload (Fragmented): gets the content of the file and divides it into different parts and stores in this field. The payload formation is depicted in the Figure 4

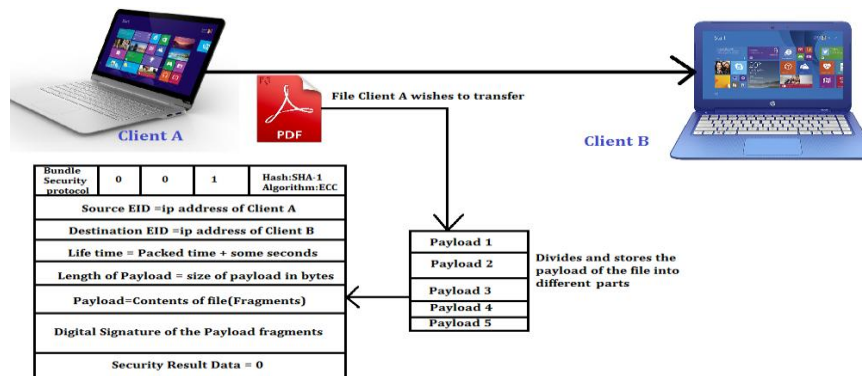


Figure 4: Bundle-Payload Formation

9. Digital Signature of payload (fragmented): Here top-and-Tail Virus Scanning method is used. Get a copy of the 'Payload' field. Each fragment is taken one by one and take the top - tail parts of it. Using any hashing algorithms such as SHA or MD5, get the hashed content of it which is now known as its Digest. Now, encrypt the Digest using the private key of Client A generated using ECC algorithm in order to get the Digital Signature of it. Do it for every other fragments and then store each Digital Signatures in this field. The Digital Signature of payload formation is depicted in the figure 5.

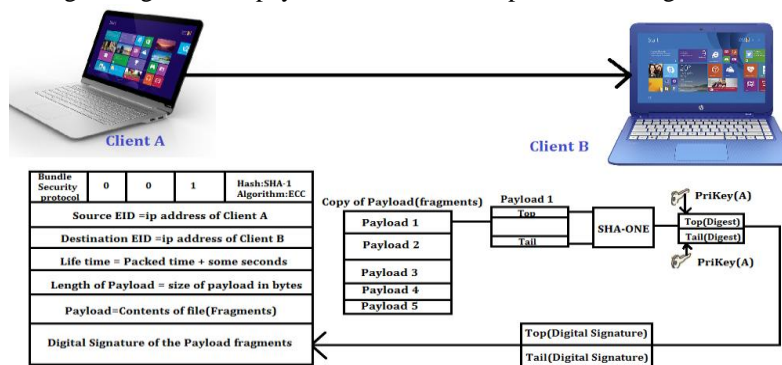


Figure 5: Bundle-Digital Signature of payload formation

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

10. Security Result Data: Initially (at the sender side) it always sets to 0. Bundle is now encrypted using the public key of the receiver and then sends it to the destination machine (here it is Client B). At last, the Bundle get encrypted using the Public Key of Client B which is shown in the Figure 6.

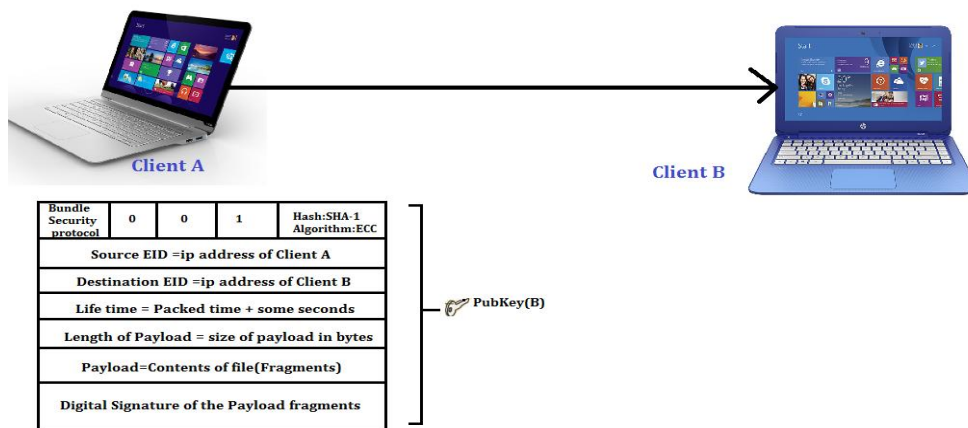


Figure 6: Encrypting the bundle

## D. File Receiving

Upon receiving the bundle from the source (Client A), recipient machine (client B) decrypts the Bundle using its own Private Key. It then extracts and check each field in the bundle. Now, it checks 'Type' field whether it contains the string "Bundle Security Protocol" or not. If true, it go for 'Bundle\_frag\_status' in the 'Processing Flag' field.If that field contains '1', it left the other sub flags and suddenly migrates to Source\_EID field as well as Destination\_EID and checks for the validity of them and sets the sub flag 'Bundle\_proc\_status' of the 'Processing Flag' field to 1.Else to 0.

It then moves on the next field called the 'lifetime'. Receiver compares the current time of the recipient machine against the value inside the 'lifetime' field. If the current time exceeds the lifetime, then set the sub flag field 'Bundle\_life\_status' of 'Processing Flag' to 0 and request the sender to resend the packet. Else, set it to 1.Then now proceed to extract the value of next fields called, 'Length of Payload', 'Payload' (Fragmented).Now, get the length of the 'payload (fragmented)' in the destination machine. Compare it against the 'length of payload' field. If they are not equal, then set 'Bundle\_proc\_status' to 0, and request the sender to resend the packet. Else, go for the Top-and-Tail Virus scanning technique.

Recipient now extracts the, 'Digital Signature of Payload' (fragmented) [DSP] field and takes the 'Payload' field too. From the DSP, recipient takes each digital signature, extract the Digests using the public key of the receiver. Now, take the corresponding top-tail fragmented parts of the field 'Payload' and do hashing on the Payload top-tail fragments in order to get the Digests using the specified hashing algorithm in the field 'Cipher Suit'.

Compare the value of Digests from the 'Payload' as well as from the 'Digital Signature'. If they are equal then set Security Result Data field to 1, else to 0.Do it for every fragments. At the end, extract the security Result data value. If it is 1, Bundle\_proc\_status sets to '1' and then the packet get accepted. Else, set to 0, and then discard the Bundle. Therefore we can claim for Confidentiality, Integrity and Availability through this method. This process is shown in the Figure 7.



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

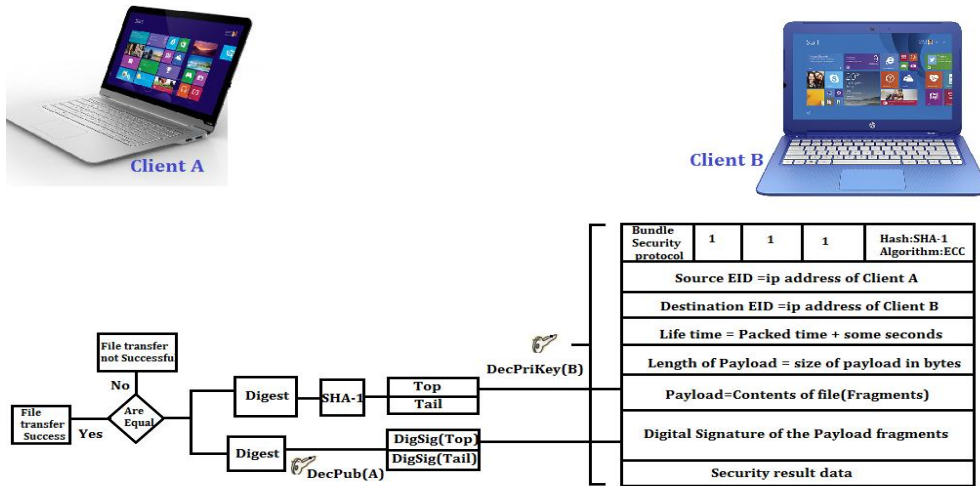


Figure 7: Decryption and Verification of Bundle

## IV. EXPERIMENTAL RESULTS

The new scheme called "Effective Transmission of Bundle in Delay Tolerant Networks" has been designed. It can effectively detect virus injection in a file and therefore assure confidentiality, integrity and availability using Bundle Security Protocol. Conducted an experiment: with the self-developed tool called SDTN (Secured Delay Tolerant Network). We have installed and tested the SDTN program in Acer PC and HP. SDTN has been written and developed in Java 7.0 using NetBeans IDE 7.0.1.

In order to test the capability of detecting virus, we have installed SDTN tool in those machines which are having JVM (Java Virtual Machine) capability. We have installed SDTN in a machine with ip address 192.168.43.94. The Figure 8 shows the other active clients in a Delay Tolerant Network with their IP address.

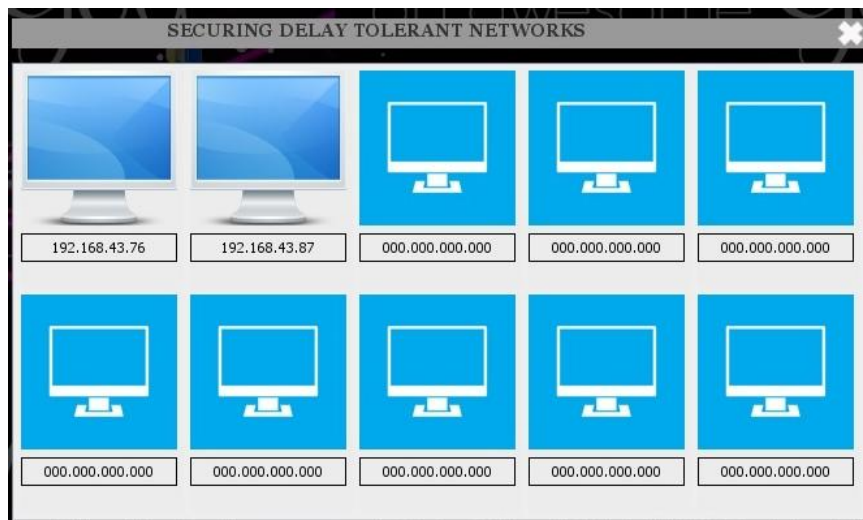


Figure 8: Active Clients in SDTN

If sender with ip 192.168.43.94 want to send a file to the client with ip address 192.168.43.87, just click on it. A sub frame called 'options' will be displayed with three buttons called, Browse File, Send File, Inbox is shown in the Figure 9.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

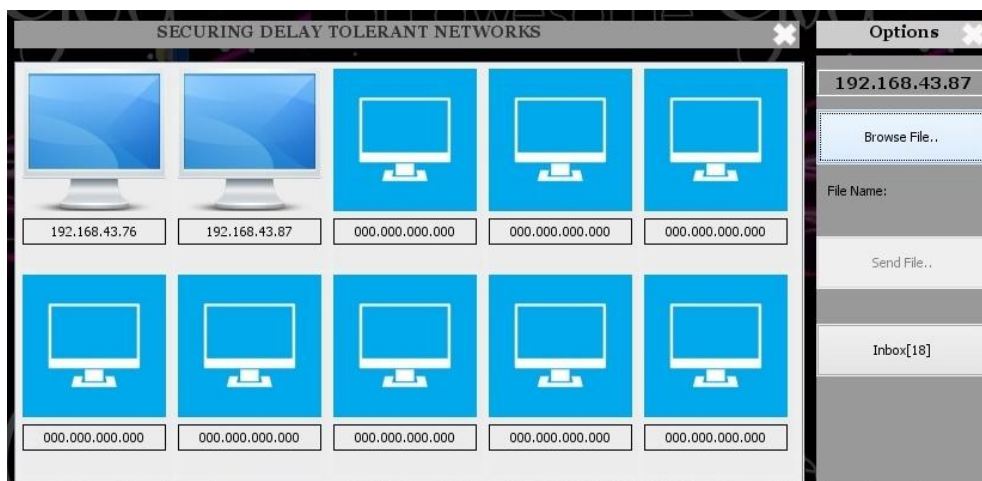


Figure 9: Sub Frame for sending a file.

Send File button will be deactivated until it selects a file. Now, it can select a file that it wishes to send using the button Browse File is shown in the Figure 10.

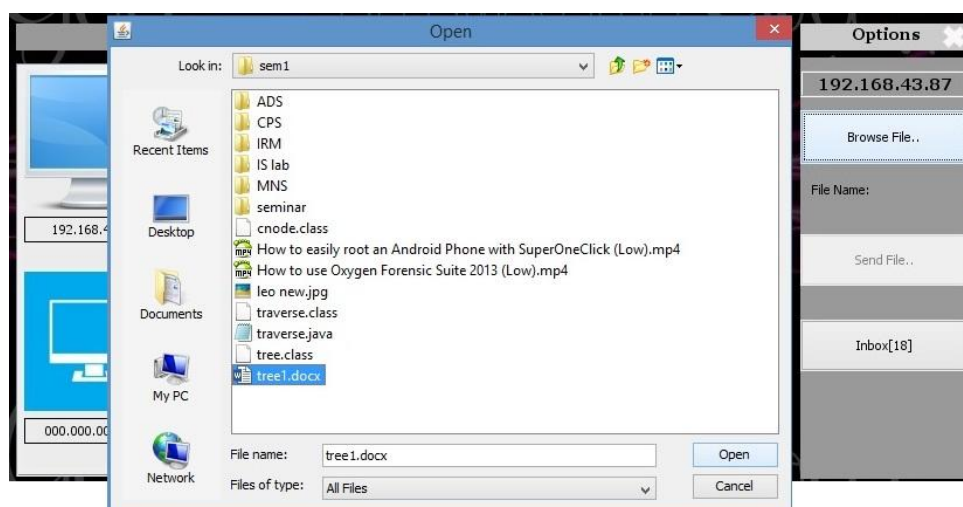


Figure 10: File browsing

After the selection, press the button Send File. At this time as explained in the proposed system, Bundle will be formed and transfers the file to the destination machine.

If the file has been reached securely, a message "File has been successfully reached" will be displayed at the destination machine. Otherwise, the message "File has been corrupted" will be displayed and automatically tells the sender to resend the Bundle. Inbox panel will be automatically displayed at the destination machine which shows the list of received files as shown in the Figure 11.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2015

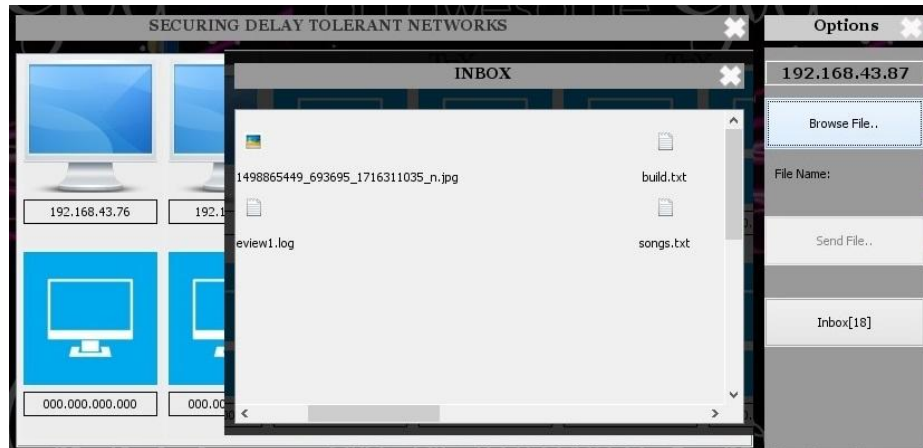


Figure 11: Inbox Panel

The tool for detecting virus and well as providing secure data transfer called the SDTN has been also successfully tested on Acer and HP and find it as efficient for detecting, preventing viruses and secure data transfers. This scheme cause lesser communication overhead and sends the file without any delays and therefore assures confidentiality, integrity and availability and also it is a solution against attacks such as DoS, Man-In-The-Middle etc.

## V. CONCLUSION AND FUTURE WORK

Delay-tolerant networking (DTN) is an approach to computer network that solves the technical issues in heterogeneous network that face the unpredictable loss of continuous network connectivity, long delay, high error rates. The new application called Secured Delay Tolerant Networks (SDTN) has been designed for securing Delay Tolerant Networks. It provides security for the data sent among the nodes in the DTN. Pattern matching of viruses is an effective method in detecting viruses. Both the pattern matching method for detecting viruses and using Bundle Security Protocol gives more strengthen DTN. This scheme is designed in such a way that it causes lesser communication overhead. Application Secured Delay Tolerant Networks, is not able to detect the viruses which are obfuscated. As a part of my future work, I would like to incorporate the method to find the obfuscated viruses as well as to design more strengthen

## REFERENCES

1. Evgenios Konstantinou Supervisor: Dr. Stephen Wolthusen Metamorphic Virus: Analysis and Detection Department of Mathematics Royal Holloway, University of London Egham, Surrey TW20 0EX, England <http://www.rhul.ac.uk/mathematics/techreports>
2. S. Symington, S. Farrell, H. Weiss, "Bundle Security Protocol Specification" Trinity College Dublin Category: Experimental, ISSN: 2070-1721
3. Pirzada, Sohail Abbas, Madjid Merabti, and David Llewellyn-Jones, "Reputation System in MANETs", <http://www.cms.livjm.ac.uk/pgnet2010/makecd/papers/2010046.pdf>.
4. Kejun Liu and Jing Deng, "Reputation values constructed using the ACK messages sent by the destination node", <http://dl.acm.org/citation.cfm?id=986876>.
5. K. Aberer and Z. Despotovic, "work on the use of reputation systems for P2P networks", <http://dl.acm.org/citation.cfm?id=776346>.
6. D. Shama and A. Kush, "GPS Enabled Energy Efficient Routing for Manet", International Journal of Computer Networks (IJCN), Vol.3, Issue 3, pp. 159-166, 2011.
7. Vellambi and F. Fekri, "challenges of providing secure communication in DTNs", in proc., International Journal of Advanced Computational Engineering and Networking 2012.
8. J. Su, K. Chan, A. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel, "A preliminary investigation of worm infections in a bluetooth environment," in Proc. ACM WORM, 2006.
9. Naveed Ahmad, Haitham Cruickshank, Zhili Sun, "ID Based Cryptography and Anonymity in Delay/Disruption Tolerant Networks", Centre for Communication Systems Research, University of Surrey Guildford, Surrey, UK





# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 10, October 2015**

9. Leona Antony, Asst.prof.Harlay Maria Mathew, Prof.P.Jayakumar: "A New Steganographic Approach Using Sudoku with Digital Signature", International Journal of Computer Engineering and Technology (IJCET) Volume 5, Issue 12, 2014, ISSN 0976-6367 (PRINT), ISSN 0976-6375 (ONLINE)
10. Y. Li, P. Hui, L. Su, D. Jin, and L. Zeng, "An optimal distributed malware defense system for mobile networks with heterogeneous devices," in Proc. IEEE SECON, 2011.
11. Ulrich Bayer, Paolo Milani Comaretti, Clemens Hlauschek, Christopher Kruegel and Engin Kirda, "Scalable, Behavior-Based Malware Clustering" in proc IEEE 2008. [12] W. Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in Proc. IEEE INFOCOM, 2007.
12. E. Daly and M. Haahr, "Social network analysis for information flow in disconnected delay-tolerant MANETs," IEEE TMC, vol. 8, no. 5, pp. 606–621, 2009.
13. S. Kamvar, M. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in P2P networks." in Proc. ACM WWW, 2003.
14. S. Buchegger and J. Boudec, "Performance analysis of the CONFIDANT protocol," in Proc. ACM MobiHoc, 2002.
15. S. Cheng, W. Ao, P. Chen, and K. Chen, "On modeling malware propagation in generalized social networks," IEEE Comm. Lett., vol. 15, no. 1, pp. 25–27, 2011.
16. Y. Li, P. Hui, L. Su, D. Jin, and L. Zeng, "An optimal distributed malware defense system for mobile networks with heterogeneous devices," in Proc. IEEE SECON, 2011.

## BIOGRAPHY

**Leona Antony** received Master's degree in Cyber Security, from Sree Narayana Gurukulam College of Engineering, Mahatma Gandhi University, Kerala, India. She received Bachelor's degree in Computer Science and Engineering in 2013 from Toc H Institute of Science and Technology, Cochin University of Science and Technology, Kerala, India.

**Litty Antony** received Master's degree in Cyber Security, from Sree Narayana Gurukulam College of Engineering, Mahatma Gandhi University, Kerala, India. She received Bachelor's degree in Information Technology in 2013 from Toc H Institute of Science and Technology, Cochin University of Science and Technology, Kerala, India.