International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-6, August 2013

Security on Dynamic Source Routing Protocol Using Onion Routing Encryption

Ritu Aggarwal

Abstract-Security in mobile ad hoc networks (MANET) is difficult to achieve, notably because of the vulnerability of wireless links, the limited physical protection of nodes, the dynamically changing topology, the absence of a certification authority, and the lack of a centralized monitoring or management point. In this paper, we embed an efficient asymmetric encryption strategy to protect and ensure anonymity for source routes when employing a source routing protocol. The base protocol used for source routing is DSR and to prevent DoS attack which occurs by modifying source route an onion based asymmetric key approach is embedded.

Keywords: DSR, DOS Attack, MANET, Onion.

I. INTRODUCTION

A mobile ad hoc network [MANET] is a temporary infrastructure less network, formed by set of wireless mobile hosts that dynamically establish their own network on the fly, without relying on any central administration [1]. Mobile hosts used in MANETs must ensure the roles that are ensured by the powerful fixed infrastructure in traditional networks. This is a challenging task, since these devices have limited resources (CPU, storage, energy, etc.). Moreover, the network's environment has some features that add extra complications, such as the frequent topology changes caused by nodes' mobility, as well as the unreliability and the bandwidth limitation of wireless channels. The security of communication in ad hoc wireless network is very important, especially in military applications. The lack of any central coordination and shared wireless medium makes them more vulnerable are generally classified into two types: passive and active attacks. Passive attacks refer to the attempts made by malicious nodes to perceive the nature of activities and to obtain information transacted in the network without disrupting the operations[10]. Active attacks disrupt the operations of the network. Those active attacks that are executed by nodes outside the network are called external attacks, and those that are performed by nodes belonging make it much more difficult to keep its security as compared to the infrastructure based networks [3].

1.1 Routing Protocols

A routing protocol is needed whenever a packet needs to be transmitted to a destination via number of nodes and numerous routing protocols have been proposed for such kind of ad hoc networks[6]. These protocols find a routefor packet delivery and deliver the packet to the correct destination. The studies on various aspects of routing protocols have been an active area of research for many years. Many protocols have been suggested keeping applications be broadly classified into two types as (a) Table Driven Protocols or Proactive Protocols and (b) On-Demand Protocols or Reactive Protocols[9].

1.1.1 Table Driven or Proactive Protocols:

In Table Driven routing protocols each node maintains one or more tables containing routing information to every other node in the network. All nodes keep on updating these tables to maintain latest view of the network. Some of the existing table driven or proactive protocols are: DSDV, DBF, GSR, WRP and ZRP[8].

1.1.2 on Demand or Reactive Protocols:

In these protocols, routes are created as and when required. When a transmission occurs from source to destination, it invokes the route discovery procedure. The route remains valid till destination is achieved or until the route is no longer needed. Some of the existing on demand routing protocols are: DSR, AODV and TORA[8][9]..

1.1.3 Dynamic Source Routing (DSR) Protocol

This is an On-demand source routing protocol. In DSR the route paths are discovered after source sends a packet to a destination node in the ad-hoc network. The source node initially does not have a path to the destination when the first packet is sent. The mobile nodes are required to maintain route caches the source routes of which the mobile is entries in the route cache are continually updated which store the complete list of IP addresses of the nodes along the path towards the destination. DSR is a source routing protocol, i.e. the complete route is given in the header of each packet. The basic procedure of DSR[4].

***A. Route discovery*:**

If the source route entry towards a destination is not present in the route cache, a Route Request packet is broadcast throughout the MANET. Before the intermediate node forwards the packet, it appends its own IP address in a list in the request packet. When the destination receives the packet, the request packet has accumulated the path from the source to the destination. Then the destination performs another route discovery to find the route towards the source if the underlying MAC layer supports unidirectional links; otherwise, it just reverses the source route recorded in the request packet. In either way, a Route Reply packet which contains the route from the source to destination is sent back to the source. After the procedure of route discovery, both the source and destination have the source route towards each other[8].

B. Route maintenance:

Unlike proactive routing protocols and AODV mentioned below, no periodic HELLO message is introduced in DSR. Every node along the path is responsible for the validity of the downstream link connecting itself and the next hop in the source route, which could be detected by MAC layer or DSR specific software acknowledgement. If link breakage is found, the source of the route will be notified with a Route Error packet.

Manuscript published on 30 August 2013.
* Correspondence Author (s)

Ritu Aggarwal, Computer Science and Enggineering, Mahrishi Markendeshwar University/ Mahrishi Markendeshwar Engineering College/ Mmu, Mullana, Jagadhri,

The source then re-initiates a route discovery procedure. Route cache is widely adopted in DSR. For example, the intermediate nodes cache the route towards the destination and backward to the source.

C. Benefits and Limitations of DSR

The main benefit of DSR protocol is that there is no need to keep routing table so as to route a given data packet as the entire route is contained in the packet header. The limitations of DSR protocol is that this is not scalable to large networks and even requires significantly more processing resources than most other protocols. Basically, In order to obtain the routing information, each node must spend lot of time to process any control data it receives, even if it is not the intended recipient. In DSR contain all information in header that’s why DOS attack is occur .for security purpose I have embedding onion routing encryption to prevent route modification attacks and check availability and integrity [4].

II. DENIAL OF SERVICE ATTACK

Denial of service (DoS) is another type of attack, where the attacker injects a large amount of junk packets into the network. These packets overspend a significant portion of network resources, and introduce wireless channel contention and network contention in the MANET. routing table overflow attack and sleep deprivation attack are two other types of the DoS attacks. In the routing table overflow attack, an attacker attempts to create routes to nonexistent nodes. Meanwhile the sleep deprivation attack aims to consume the batteries of a victim node. For example, consider the following .Assume a shortest path exists from **S** to **X** and **C** and **X** cannot hear each other, that nodes **B** and **C** cannot hear each other, and that **M** is a malicious node attempting a denial of service attack [2].

Suppose **S** wishes to communicate with **X** and that **S** has an unexpired route to **X** in its route cache. **S** transmits a data packet toward **X** with the source route **S** --> **A** --> **B** --> **M** --> **C** --> **D** --> **X** contained in the packet’s header. When **M** receives the packet, it can alter the source route in the packet’s header, such as deleting **D** from the source route. Consequently, when **C** receives the altered packet, it attempts to forward the packet to **X**. Since **X** cannot hear **C**, the transmission is unsuccessful.

S ↔A↔ B↔ M ↔C↔ D↔ X

Figure 1Denial of Service

III. RELATED WORKS

Michael Backes [3] presenting a security definition (an ideal functionality) for the OR methodology in the universal composability (UC) framework. We then determine the exact security properties required for OR cryptographic primitives (onion construction and processing algorithms, and a key exchange protocol) to achieve a provably secure OR protocol. We show that the currently deployed onion algorithms with slightly strengthened integrity properties can be used in a provably secure OR construction. In the process, we identify the concept of predictably malleable symmetric encryptions, which might be of independent interest. On the other hand, we find the currently deployed key exchange protocol to be inefficient and difficult to analyze and instead show that a recent, significantly more efficient, key exchange protocol can be used in a provably secure OR construction. In addition, our definition greatly simplifies the process of analyzing OR anonymity metrics. We define and prove Forward secrecy for the OR protocol, and realize our (white box) OR definition from an OR black-box model assumed in a recent anonymity analysis. This realization not only makes the analysis formally applicable to the OR protocol but also identifies the exact adversary and network assumptions made by the black box model.

E. Mohammadi ,Goldberg, Stebila and Ustaoglu [3] the onion routing (OR) network Tor provides privacy to Internet users by facilitating anonymous web browsing. It achieves anonymity by routing encrypted traffic across a few routers, where the required encryption keys are established using a key exchange protocol. Goldberg, Stebila and Ustaoglu recently characterized the security and privacy properties required by the key exchange protocol used in the OR network. They defined the concept of one-way authenticated key exchange (1W-AKE) and presented a provably secure 1W-AKE protocol called ntor, which is under consideration for deployment in TOR. In this paper, we present a novel 1W-AKE protocol Ace that improves on the computation costs of ntor: in numbers, the client has an efficiency improvement of 46% and the server of nearly 19%. As far as communication costs are concerned, our protocol requires a client to send on additional group element to a server, compared to the ntor protocol. However, an additional group element easily fits into the 512 bytes fix-sized Tor packets (or cell) in the elliptic curve cryptography (ECC) setting. Consequently, our does not produce a communication overhead in the Tor protocol. Moreover, we prove that our protocol Ace constitutes a 1W-AKE. Given that the ECC setting is under consideration for the Tor system, the improved computational efficiency, and the proven security properties make our 1W-AKE an ideal candidate for use in the Tor protocol.

IV. PROPOSED WORK

This work is about to secure the dedicated route and detect the Malicious Attack if any in case of MANET. In this research we are presenting the complete work with onion routing and encryption using DSR Protocol. It analysis of existing methodologies to secure the whole network and defend MANET against the route modification attack/dos attacks. According to this approach if a source node wants to a send the packet to destination node. Only the source node knows the address of its successor node, not to next, next node. Like a hidden technique, or like a when we peel of onion layers[4]. Then we present the mechanism by using asymmetric cryptography using RSA algorithm. In the following discussion we assume that the initiator S performs a route discovery for the target D, and that have own private key for decryption and encrypt using individuals public key they are different, respectively the detect the DOS attacks. The complete work is defined in terms of some stages .Which kind of Network can be used to perform the communication? How many nodes are sufficient to define the proposed work? Which cryptographic algorithm will be used to match the used? How the problem will be resolved.

Which environment should be implementing to find the solution of the defined problem? Using DSR Routing Protocol to prevent the route modification attack /DOS attacks using RSA algorithm implementation. To find the route between different node. Using onion routing encryption for securing the route by using asymmetric cryptography. How the result will be analyzed as the works begin with defining the answers of the above said questions and a complex research solution is obtained that provide us a reliable communication over the network.

4.1Embedding Onion Routing in DSR to Prevent DOS Attack

In Onion Routing purpose the use of an efficient asymmetric encryption strategy (private key and public key) to protect and ensure anonymity for source routes when employing a source routing protocol. This strategy consists of encrypting a discovered source route during route discovery in an *onion-like* form, and transmitting data packets using this onion encrypted route. During the route reply (respectively request broadcasting) phase, each node adds its address to the next (respectively previous) portion of the discovered route, and encrypts the outcome using the public key of the previous node (respectively its own public key). In this manner each node will be able to only read the next hop when data packets are transmitted, and not any other. The onion encryption of a discovered source route (*n*0,*n*1, …, *nk*) is performed during the reply phase as follows: *nk* ID is encrypted with *Pnk*–1 (the public key of node *nk*–1),the result is denoted by [*nk*]*Pnk*–1; at *nk*–l this outcome is concatenated to *nk*–1 ID and encrypted with *Pnk*–2: [*nk*–1,[*nk*]*Pnk*–1]*Pnk*–2, and so on until reaching the source’s successor(*n*1). The outcome of all these operations is the following onion encrypted source route: [*n*1, …, [*nk–1*, [*nk*]*Pnk*–1]*Pnk*–2 …]*Pn*0. This encrypted route will be used to route each data packet. Node *n*0 decrypts the route and gets *n*1 address, to which it transmits the packet; the remaining part is encrypted with *pn*1 and cannot be deciphered by *n*0. *n*1 does the same thing and routes the packet, and so on until reaching the final destination. Assume a discovered source route (B, C, D), which connects A to D, is to be used by A to transmit a data packet [9].The onion-encrypted sequence of this route is: [*B*, [*C* [*D*]*PC*] *PB*]*PA*. When decrypting the route with its own private key, node A retrieves B’s address, to which it transmits the packet. The other addresses (C, D) are hidden to A, and cannot be deducted since they are asymmetrically encrypted (assuming the asymmetric encryption mechanism is robust). Identically, B (respectively C) gets C’s (respectively D’s) address, using its own private key, to which it forwards the packet. This mechanism ensures that each node is only able to identify its successor, where the rest of the route is kept anonymous. Consequently, DoS attack by modifying source route is prevented. When combined with authentication, this mechanism is powerful and efficient, but it suffers from high computation cost[4][5].

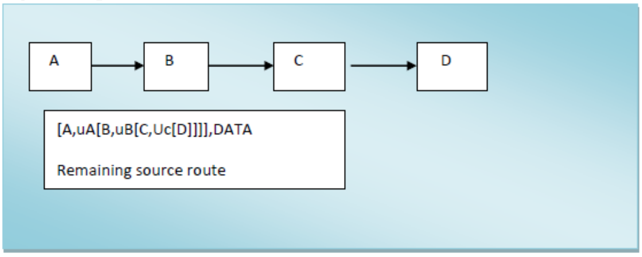


Fig2 forwrading a packet using onion routing encryption

V. SYSTEM MODEL

Scenarios With Framework Of Nodes : The MANET’s comprises of frame of nodes. The Output shown that when basic network frame, and Node frame structure.

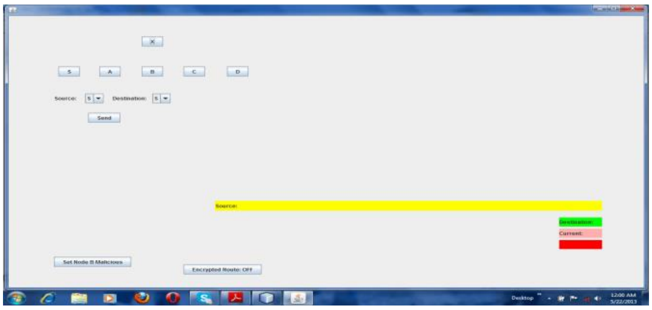
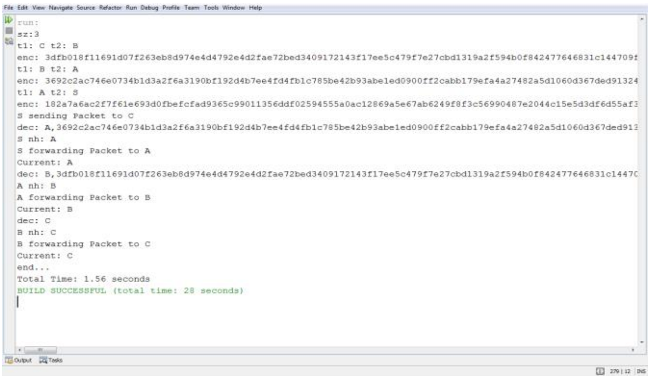


Figure3 framework of nodes

5.1 Simulation Scenarios With Framework Of 3 Mobile Nodes

The MANET’s comprises of 3 mobile nodes. Where the source S is sending the packet to destination node C and source uses own private key to decrypt the node A packet and A has own public key. Then node A using his private key to decrypt the node B packet and so on. This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 1.56 seconds


File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output
Tasks
279 | 12
INS

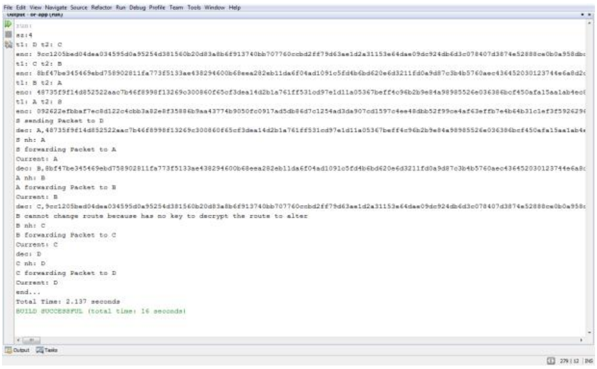
```
run:
sz:3
t1: C t2: B
enc: 3dfb018f11691d07f263eb8d974e4d4792e4d2fae72bed3409172143f17ee5c479f7e27cbd1319a2f594b0f842477646831c144709f
t1: B t2: A
enc: 3692c2ac746e0734b1d3a2f6a3190bf192d4b7ee4fd4fb1c785be42b93abe1ed0900ff2cabb179efa4a27482a5d1060d367ded91324
t1: A t2: S
enc: 182a7a6ac2f7f61e693d0fbefcfad9365c99011356ddf02594555a0ac12869a5e67ab6249f8f3c56990487e2044c15e5d3df6d55af3
S sending Packet to C
dec: A,3692c2ac746e0734b1d3a2f6a3190bf192d4b7ee4fd4fb1c785be42b93abe1ed0900ff2cabb179efa4a27482a5d1060d367ded913
S nh: A
S forwarding Packet to A
Current: A
dec: B,3dfb018f11691d07f263eb8d974e4d4792e4d2fae72bed3409172143f17ee5c479f7e27cbd1319a2f594b0f842477646831c1447C
A nh: B
A forwarding Packet to B
Current: B
dec: C
B nh: C
B forwarding Packet to C
Current: C
end...
Total Time: 1.56 seconds
BUILD SUCCESSFUL (total time: 28 seconds)
```

Figure4 Simulation Scenarios with the 3 Mobile Nodes

5.2 Simulation Scenarios With When Malicious Activity Is Attack

The MANET’s comprises of 4 mobile nodes. Where the source S is sending the packet to destination node D and source uses own private key to decrypt the node A packet and A has own public key. Then node B has act as an attacker but due to onion encryption it cannot change route because has no key to decrypt the route to alter, This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 2.317 seconds.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.



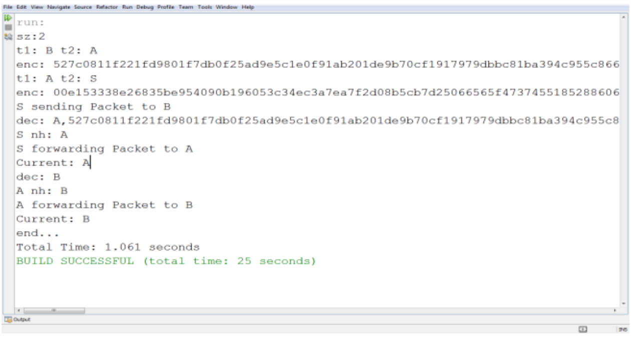
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output - or-app (run)
Output Tasks
279 | 12 | INS

```
run:
sz:4
t1: D t2: C
enc: 9cc1205bed04dea034595d0a95254d381560b20d83a8b6f913740bb707760ccbd2ff79d63ae1d2a31153e64dae09dc924db6d3c078407d3874e52888ce0b0a958dbc
t1: C t2: B
enc: 8bf47be345469ebd758902811fa773f5133ae438294600b68eea282eb11da6f04ad1091c5fd4b6bd620e6d3211fd0a9d87c3b4b5760aec43645203012374 4e6a8d2c
t1: B t2: A
enc: 48735f9f14d852522aac7b46f8998f13269c300860f65cf3dea14d2b1a761ff531cd97e1d11a05367beff4c96b2b9e84a98985526e036386bcf450afa15aa1ab4ec8
t1: A t2: S
enc: 092622efbbaf7ec8d122c4cbb3a82e8f35886b9aa43774b9050fc0917ad5db86d7c1254ad3da907cd1597c4ee48dbb52f99ce4af63effb7e4b64b31c1ef3f592629
S sending Packet to D
dec: A,48735f9f14d852522aac7b46f8998f13269c300860f65cf3dea14d2b1a761ff531cd97e1d11a05367beff4c96b2b9e84a98985526e036386bcf450afa15aa1ab4e
S nh: A
S forwarding Packet to A
Current: A
dec: B,8bf47be345469ebd758902811fa773f5133ae438294600b68eea282eb11da6f04ad1091c5fd4b6bd620e6d3211fd0a9d87c3b4b5760aec43645203012374 4e6a8c
A nh: B
A forwarding Packet to B
Current: B
dec: C,9cc1205bed04dea034595d0a95254d381560b20d83a8b6f913740bb707760ccbd2ff79d63ae1d2a31153e64dae09dc924db6d3c078407d3874e52888ce0b0a958c
B cannot change route because has no key to decrypt the route to alter
B nh: C
B forwarding Packet to C
Current: C
dec: D
C nh: D
C forwarding Packet to D
Current: D
end...
Total Time: 2.137 seconds
BUILD SUCCESSFUL (total time: 16 seconds)
```

Figure5 Simulation Scenarios with the 4 Mobile Nodes

5.3 Simulation Scenarios With Framework Of 2 Mobile Nodes

The MANET's comprises of 2 mobile nodes. Where the source S is sending the packet to destination node B and source uses own private key to decrypt the node A packet and A has own public key. Then node A using his private key to decrypt the node B packet and so on. This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 1.03 seconds.



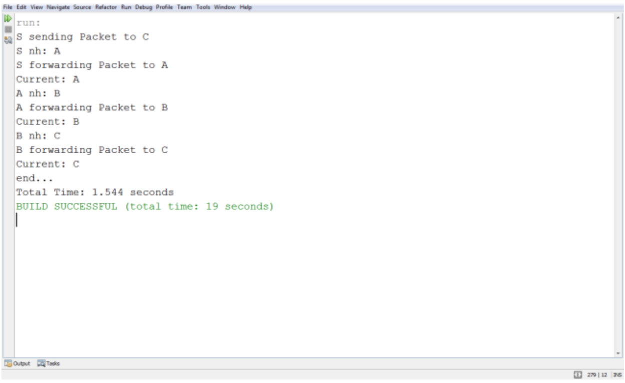
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output
INS

```
run:
sz:2
t1: B t2: A
enc: 527c0811f221fd9801f7db0f25ad9e5c1e0f91ab201de9b70cf1917979dbbc81ba394c955c866
t1: A t2: S
enc: 00e153338e26835be954090b196053c34ec3a7ea7f2d08b5cb7d25066565f47374551852886 06
S sending Packet to B
dec: A,527c0811f221fd9801f7db0f25ad9e5c1e0f91ab201de9b70cf1917979dbbc81ba394c955c8
S nh: A
S forwarding Packet to A
Current: A
dec: B
A nh: B
A forwarding Packet to B
Current: B
end...
Total Time: 1.061 seconds
BUILD SUCCESSFUL (total time: 25 seconds)
```

Figure6 Simulation Scenarios with the 2 Mobile Nodes

5.4 Simulation Scenarios With Framework Of 3 Mobile Nodes Without Encrypted Route

The MANET's comprises of 3 mobile nodes. Where the source S is sending the packet to destination node C and source uses own private key to decrypt the node A packet and A has own public key. Then node A using his private key to decrypt the node B packet and so on. This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 1.544 seconds.



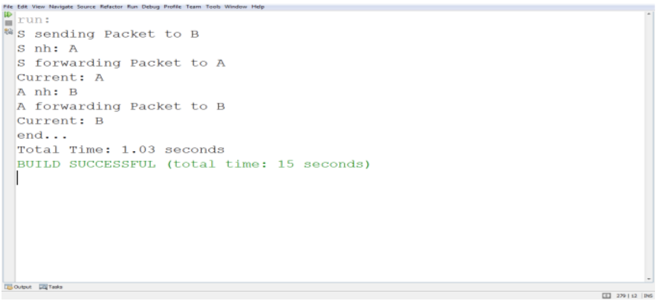
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output Tasks
279 | 12 | INS

```
run:
S sending Packet to C
S nh: A
S forwarding Packet to A
Current: A
A nh: B
A forwarding Packet to B
Current: B
B nh: C
B forwarding Packet to C
Current: C
end...
Total Time: 1.544 seconds
BUILD SUCCESSFUL (total time: 19 seconds)
```

Figure7 Simulation Scenarios with the 3 Mobile Nodes

5.5 Simulation Scenarios with Framework Of 2 Mobile Nodes

The MANET's comprises of 2 mobile nodes. Where the source S is sending the packet to destination node B and source uses own private key to decrypt the node A packet and A has own public key. Then node A using his private key to decrypt the node B packet and so on. This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 1.03 seconds.



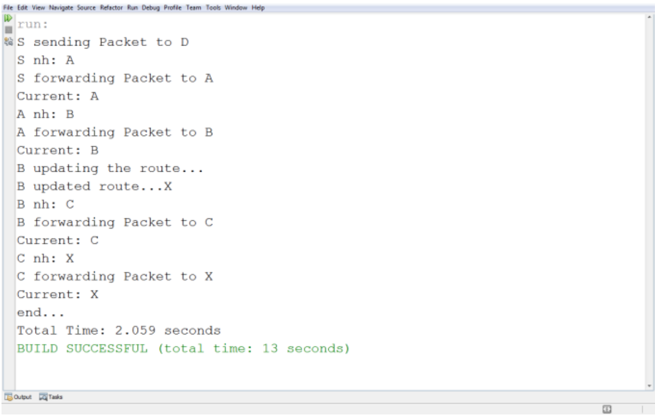
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output Tasks
279 | 12 | INS

```
run:
S sending Packet to B
S nh: A
S forwarding Packet to A
Current: A
A nh: B
A forwarding Packet to B
Current: B
end...
Total Time: 1.03 seconds
BUILD SUCCESSFUL (total time: 15 seconds)
```

Figure8 Simulation Scenarios with the 2 Mobile Nodes

5.6 Simulation Scenarios With Framework Of 4 Mobile Nodes When Encrypted Route Is Off And Malicious Activity Attack

The MANET's comprises of 3 mobile nodes. Where the source S is sending the packet to destination node D and source uses own private key to decrypt the node A packet and A has own public key. Then node B has act as an attacker node B updating the route to node X. This simulation is done when we implemented Onion Routing Encryption time taken for sending and receiving the packet is 2.059 seconds.

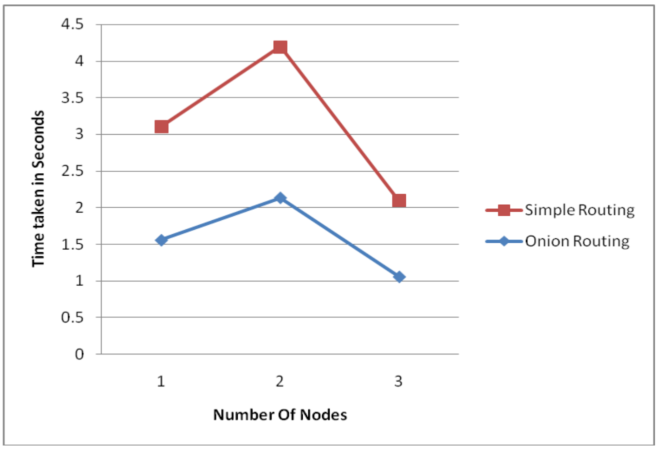


File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Output Tasks

```
run:
S sending Packet to D
S nh: A
S forwarding Packet to A
Current: A
A nh: B
A forwarding Packet to B
Current: B
B updating the route...
B updated route...X
B nh: C
B forwarding Packet to C
Current: C
C nh: X
C forwarding Packet to X
Current: X
end...
Total Time: 2.059 seconds
BUILD SUCCESSFUL (total time: 13 seconds)
```

Figure9 Simulation Scenarios with the 4 Mobile Nodes

5.7 Result Analysis



Above shown in graph obtained from the Simulation of our implemented ONION Routing. As we can see for Onion routing time taken for the dedicate route from source to destination is for 3 mobile nodes is 1.56 seconds, when attacker attacks the route time taken for 4 mobile nodes is 2.137 seconds, with the 2 mobile nodes the time taken is 1.061 seconds. As we can see in Simple routing time taken for the dedicate route from source to destination for 3 mobile nodes is 1.544 seconds, when attacker attacks the route time taken for 4 mobile nodes is 2.059 seconds, with the 2 mobile nodes the time taken is 1.03 seconds.

VI. ACKNOWLEDGEMENT

We wish to express our deep sense of indebtedness and sincerest gratitude to our guide, **Dr Rajneesh Gujral**, Department of Computer Engineering, MMEC, MULLANA (Ambala), India. providing me with exactly the amount of structure needed to ensure my success. You have not simply taught me how to succeed as a student, but rather how to be an independent researcher. Thank you so much for all of the academic, professional, and personal advice that you have given me.

REFERENCES

[1]. *Jeoren Hoebeke, Ingrid Moerman, Bart Dhoedt and Piet Demester* "An Overview of Mobile ad hoc Networks: Applications & Challenges.

[2]. *Josh Broch, David B. Johnson, and David A. Maltz*. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. Internet-Draft, *draft-ietf-manet-dsr-03.txt, October 1999*. Work in progress.

[3]. *M. Backes, I. Goldberg, A. Kate, and E. Mohammadi*,"Provably secure and practical onion routing," IACR Cryptology ePrint Archive, *Report 2011/308, 2012.*

[4]. *W. Stallings, Cryptography and Network Security Principles and Practices*, 3rd ed., Pearson Education Inc., 2003.

[5]. *D. B. Johnson and D. A. Maltz*, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers, *1996,vol. 353, ch. 5, pp. 153–181.*

[6]. *Quan Jia, Kun Sun, Angelos Stavrou, "CapMan*: Capability-based Defense against Multi- Path Denial of Service (DoS) Attacks in MANET", *Proceedings of 20th International Conference on Computer Communications and Networks (ICCCN),Maui, HI, USA, 2011, July 31-August 4, 2011, pp.1-6.*

[7]. *D. B. Johnson, D. A. Maltz, Y.-C. Hu, and J. G. Jetcheva, "*The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-09.txt, Apr. 2003.*

[8]. *Charles E. Perkins,* Ad Hoc Networking", *2001.*

[9]. *Perrig, A., Canetti, R., Song, D., and J. Tyger*, "Efficient and Secure Source Authentication for Multicast", Network and Distributed System Security mposium, *NDSS 2001, pp. 35-46, February 2001.*

[10]. *Deshpande Vivek S*," Security in Ad-Hoc Routing Protocols" Pune, Maharashtra, *India 1999.*