iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 8 **Issue:** IX **Month of publication:** September 2020

**DOI:** https://doi.org/10.22214/ijraset.2020.31501

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429*
*Volume 8 Issue IX Sep 2020- Available at www.ijraset.com*

# Secure GPRS Based Routing in MANETS

Botta Venkata Kavya[1]
[1]*M. Tech Computer Science, MVRG College of Engineering, Vizianagaram, Andhra Pradesh*

***Abstract:*** ***These days key establishing of two devices in a network places an important role, and the generation of the key is used for public key-based algorithms. Using this algorithm generates a secret key for two devices in the network so that we can randomly generate private keys by performing these processes. In the Ad hoc networks, the essential operation of routing is from the source node to the destination node; by randomly generating this process, we can improve the routing efficiency. In this paper, we implement the random routing of secure transmission protocol for generating routing and provide the privacy for transferred messages among the nodes by using the GPSR protocol. Before moving the news, the server will randomly generate the routing path for the S-node and D-node. These transferring messages or data are encrypted in the s-node, and cipher is sent to the d-node, then the d-node will retrieve the cipher format data and perform the decryption process and gathers the original data or message. By implementing this concept in the nodes, we can improve the efficiency and security in generating the routing path for transferring messages.***

***Keywords:*** ***Security, Secret Key Establishment, Greedy perimeter stateless routing protocol.***

## I. INTRODUCTION

Mobile ad-hoc networks collect hundreds and thousands of low cost and low power mobile nodes connected by wireless links [1]. The operation of the MANETs is based on the routable network properties where each node acts as a router to forward the data traffic to a specified node in the network. The mobile nodes are arbitrarily organized themselves and are allowed to move freely. In the ad-hoc network, the units are usually small, portable, and battery-powered, and their communication is through radio frequency signal, [3]

Since their signal reach is limited, they can only communicate within their range of signal transmissions. These nodes can even transmit data to other nodes beyond their scope by acting as routers, forwarding the information from the S-node to D-node. The pairwise critical establishing process firstly, the user node gathers the neighbour information and then checks whether they share the same pre-distributed keys, then they establish pairwise keys between them directly. Now the nodes can generate a random key pair and passes them to its neighbour with the encrypted shared keys.

The advantage of mobile ad hoc networks is to provide access to information and services regardless of geographic area, and also it works without any infrastructure, which does not have a form of any network infrastructure for connecting between mobile nodes. This paperwork analyses the need for security and performance of the nodes that participate in a random key generation and the Greedy perimeter routing protocol to exhibit the efficiency, throughput, and reduce the data packet loss, packet delay, and increase in the packet delivery ratio. This research will also lead to the reliability of the data transmission and improve the efficiency of the transferred messages and the security for the routing paths.

MANETs builds secret common randomness between two nodes or multiple nodes in a network that resides at the root of communication security. The critical establishment is a challenging problem in the sensor networks because of the resource limitations of the nodes and vulnerability to the sensor nodes' physical capture. The existing system scheme provides a significantly better trade-off between communication overhead, computational overhead, network connectivity, and security against node capture compared to the current critical pre-distribution strategy. According to the proposed system, the packet exchange increases significantly along with the packet pickup time and packet delivery time, including the source node random key generation and destination node capturing the public key from its neighbour node and its private key distribution. This will lead to the efficiency and increased security rate for the random nodes, which help find the shortest path for the destination to transfer the data transmission.

## II. RELATED WORK

Krishna Kumar et al. (2015) proposed a secret key understanding between two or numerous gadgets in a typically needy system upon an open key framework. Be that as it may, in the situations when no such framework exists, or when the existing framework is not dependable, clients are left with generally a couple of strategies for setting up a secure correspondence. This paper discusses KERMAN, a secret common haphazardness foundation calculation for impromptu systems, which works by reaping haphazardness straightforwardly from the organize directing metadata along these lines accomplishing both unadulterated irregularity era and

(indeed) mystery key assertion. KERMAN depends on the course disclosure period of an impromptu system utilizing the Dynamic Source Routing convention. The calculation is assessed for different system parameters, and two unique levels of many-sided quality, in an OPNET automatic system test system. Our outcomes demonstrate that, in a brief span, a vast number of mystery irregular bits can be produced organize complete, between various matches in a system of fifty clients.

Ashish Khisti and Suhasi (2012) creator giving arrangement on meddler watch a source grouping related with the honest to goodness terminals. Mystery key limit is set up when the sources grouping of the meddler and the channel of the spy are debased renditions of the relating source and tracks at the true-blue recipient. At the point when an open discourse channel is accessible, propose creating separate mystery keys from sources and channels and build up its optimality in some exceptional cases. A mystery key assertion procedure that saddles vulnerabilities from both sources and media. Our lower bound rate expression includes selecting a working point that adjusts source and channel prevarications' commitment. Its optimality is built up for the instance of conversely corrupted parallel channels.

T. Perarasi and G. Nagarajan (2018) proposed a scheme about the key distribution as the central core idea, which is based on the methodology called differentiated key pre-distribution. This idea is for distributing various numbers of keys to different nodes to enhance the resilience of specific connections. Establishing end-to-end secure relationships between source and sink is significant for many Cognitive Radio Networks. These works improve that during the routing process, the user's route through those links with high resilience by the key pre-distribution factor. Using theoretical analysis, the quality of end-to-end communication is secured and uses it as protocol parameters optimally.

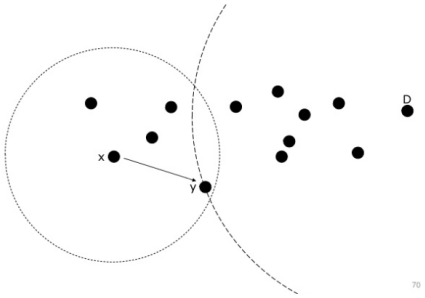
## III. PROPOSED SYSTEM

In the proposed system we are introducing the Greedy perimeter routing protocol so that it would lead to an increase in efficiency and security for the node to find the shortest path. By implementing this protocol, we can generate a secret key, generate randomness routing, encryption, and decryption of transferring messages.

### *A. Routing protocols for WSNS*

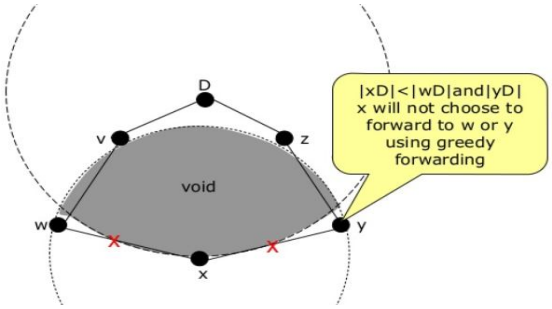
The wireless sensor networks can be processed with two main routing protocols: The Location-Centric and The Data-Centric routing. Sensor nodes are addressed employing their locations. The distance between the neighbor nodes can be estimated based on incoming signal strengths. Relative coordinates of neighbor nodes can be obtained by exchanging such information between neighbors. Some location-based schemes demand that nodes go to sleep if there is no activity to save energy. More energy savings can be obtained by having as many sleeping nodes in the network as possible. At this moment, two crucial location-based routing protocols, GEAR, and GPSR are introduced.

GPSR (Greedy Perimeter Stateless Routing): uses the packets to follow the perimeter of the planar graph to find their routes. Although the GPSR approach reduces the number of states a node should keep, it has been designed for general mobile ad-hoc networks and requires a location service to map locations and node identifiers. This algorithm consists of two methods: a) greedy forwarding and b) perimeter forwarding. Greedy forwarding, which is used wherever possible, and perimeter forwarding used in the region's greedy forwarding cannot be done. Under GPSR, packets are marked by their originator with their destination locations. As a result, a forwarding node can make a locally optimal, greedy choice to choose a next-hop packet. Especially if a node knows its radio neighbors' positions, the next hop's locally optimal choice is the neighbor geographically closest to the packet's destination. Forwarding in this scheme follows successively closer geographic bounds until the goal is reached.



*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429*
*Volume 8 Issue IX Sep 2020- Available at www.ijraset.com*

A simple beaconing algorithm provides all nodes with their neighbor's positions: periodically, each node transmits a beacon to broadcast MAC addresses, containing its identifier and status. The situation is encoded as two four-byte floating-point quantities, for x and y coordinates values; upon not receiving a beacon from a neighbor for longer than timeout interval T, a GPSR router assumes that the neighbor has failed or gone-out-of-range, and deletes the neighbor from its neighbor table. GPSR benefit all stem from geographic routings use of only immediate-neighbor information in forwarding decisions.




*B. Nodes Initiation Process*

In this module, we are generating the communication process of each node to the server. Before performing all three concepts, we develop the communication of each node. The communication process can be done by sending the IP address and port number of servers. After sending a request, the server will accept the offer and generate communication between nodes. Before performing the touch, the server will create points ($X_i$, Yi) for each node and send it to each node in a wireless sensor network. The implementation of the secret key is as follows.

*1) Secret Key Generation Process*

*a)* The source node and destination node will choose two prime numbers A and B.

*b)* The source node will enter the private key (p) and generate the public key using the following formula.

Public key = $B^P \bmod A$

*c)* After generating a public key, the source node will send it to the destination node.

*d)* The destination node will retrieve the public key, and the destination node will enter the private key (q) to calculate the public access.

Destination public key= $B^q \bmod A\,1$

*e)* The destination node will send the public key to the source node and generate a shared key using the following formula.

Shared key= destination public key $^{p}{}_{\bmod A}$

*f)* The destination node will retrieve the source node public key and generate a shared key using the following formula.

Shared key= source node pubic key $^{q}{}_{\bmod A}$

After completing this process, the source node and destination node will get the same secret key type. The completion of the private key, the source node will enter the transferred message and perform the encryption process. After moving, the server will generate routing from source node to destination node. The generation of routing can be done randomly, and the implementation is shown as follows.

*C. Route Discovery Process*

In this process, the source node will send a request to the initial node and generate random routing using the following method.

*1)* The initial node will retrieve all points of individual clients.

*2)* After getting those points, the server will determine the difference between source nodes to other nodes by using the following formula.

Diff= sqrt(X2- X1+Y2-Y1)

*3)* The calculating difference, we can generate a random path and calculating the distance of all routes by adding contrast.

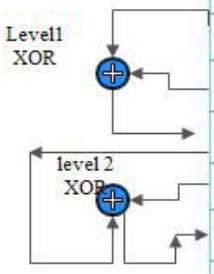
*4)* Then take the values of all routers and find out the minimum length of the path, and send the data through that path.

*5)* Before searching for the path source node will enter the message ,perform the encryption process, the implementation process is as follows.

*a) Encryption*

P= Simple-Text

- Add the randomized characters in between the simple text. For every three characters, add one same name.
- Get the binary codes for the characters in simple text.
- Do the complement of the simple text.
- Select series of prime numbers and convert them into Binary format.
- Do the first level Exclusive OR (XOR) between straight text characters and selected series of prime numbers.
- Select any Randomized number (key). Get access to prime numbers from the prime numbers table.
- Do the Second level of XOR operation between the result of step 5 and Randomized prime number.
- Convert the output of step6 into decimal values. Now you will get the ciphertext.

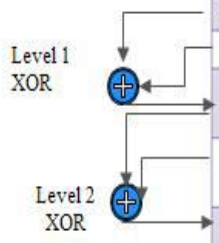



| plaintext | B | A | N | X | A | N | A | S |
|---|---|---|---|---|---|---|---|---|
| ASCII | 66 | 65 | 78 | 88 | 65 | 78 | 65 | 83 |
| Binary number | 01000010 | 01000001 | 01001110 | 1011000 | 01000001 | 01001110 | 01000001 | 1010011 |
| Complement | 10111101 | 10111110 | 10110001 | 10100111 | 10111110 | 10110001 | 10111110 | 10101100 |
| Prime numbers | 00011101 | 00011111 | 00100101 | 00101001 | 00101011 | 00101111 | 00110101 | 00111011 |
| Level 1 Result | 10100000 | 10100001 | 10010100 | 10001110 | 10010101 | 10011110 | 10001011 | 10010111 |
| KEY | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 |
| Level 2 Result | 01000101 | 01000100 | 01110001 | 01101011 | 01110000 | 01111011 | 01101110 | 01110010 |
| Cipher Text | 69 | 68 | 113 | 107 | 112 | 123 | 110 | 114 |

After completing the encryption process, the source node will send cipher format data to the destination node through a path. The destination node will gather cipher format data and perform the decryption process. The implementation process is as follows.

*b) Decryption*

- Convert the ciphertext into Binary format. Get the Key prime number from the prime numbers table. And convert it into binary form.
- Do the first level of XOR-operation between ciphertext and Key the primary-key.
- Select the series of prime numbers and convert them into the binary format (the sequence must be the same in both the encryption and decryption sides).
- Do the second level of XOR operation between the result of step2 and selected series of prime numbers.
- Get a complement to the development of step4. Then process the result from binary to decimal format.
- . Remove the randomized stuffed numbers. Now you can get the plaintext.

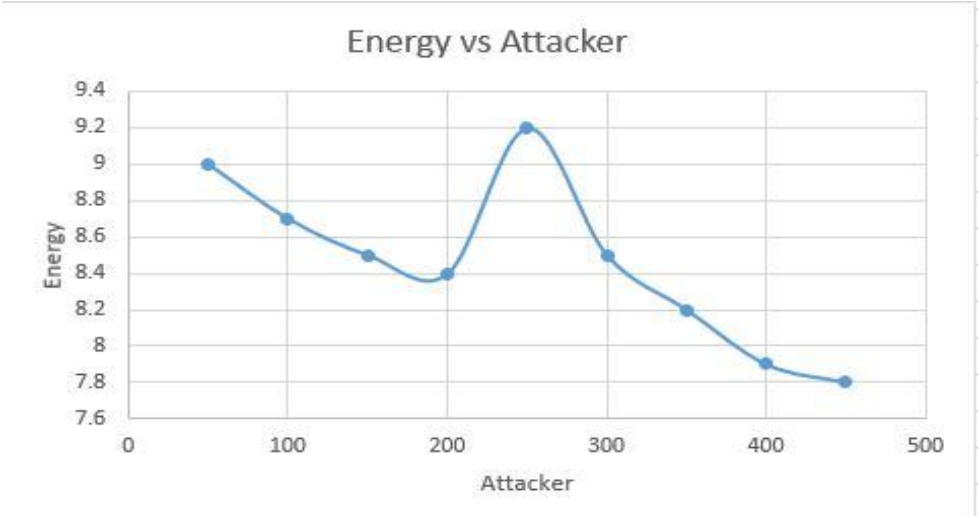



| Cipher text | 69 | 68 | 113 | 107 | 112 | 123 | 110 | 114 |
|---|---|---|---|---|---|---|---|---|
| Binary code | 01000101 | 01000100 | 01110001 | 01101011 | 01110000 | 01111011 | 01101110 | 01110010 |
| Key | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 | 11100101 |
| Level 1 Result | 10100000 | 10100001 | 10010100 | 10001110 | 10010101 | 10011110 | 10001011 | 10010111 |
| Prime Number | 00011101 | 00011111 | 00100101 | 00101001 | 00101011 | 00101111 | 00110101 | 00111011 |
| Level 2 Result | 10111101 | 10111110 | 10110001 | 10100111 | 10111110 | 10110001 | 10111110 | 10101100 |
| Complement | 01000010 | 01000001 | 01001110 | 01011000 | 01000001 | 01001110 | 01000001 | 01010011 |
| P (ASCII) | 66 | 65 | 78 | 88 | 65 | 78 | 65 | 83 |
| Plain Text | B | A | N | - | A | N | A | - |

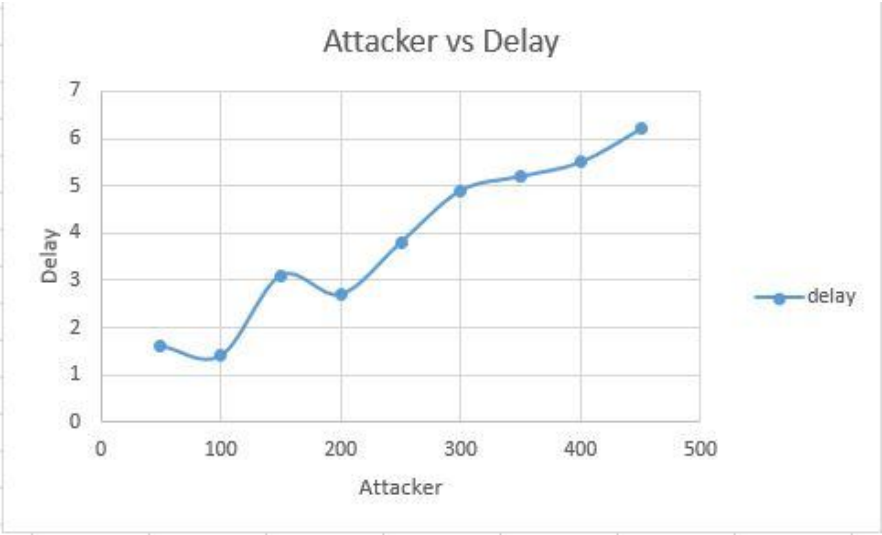
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429*
*Volume 8 Issue IX Sep 2020- Available at www.ijraset.com*

## IV. RESULT AND DISCUSSION

This scheme's impact shows the slight difference between the attackers and energy, thus displays the increase and decrease of the flow when coming to the number of nodes at which the attackers try to capture the data during the data transmission.




Here the attackers try to hack to the network nodes resulting in transmitting energy getting reduced to reach the destination node. Attackers track the destination node and attempt to interfere with the transmitter, which ensures in calculating the delay in the network depicts the above statement.

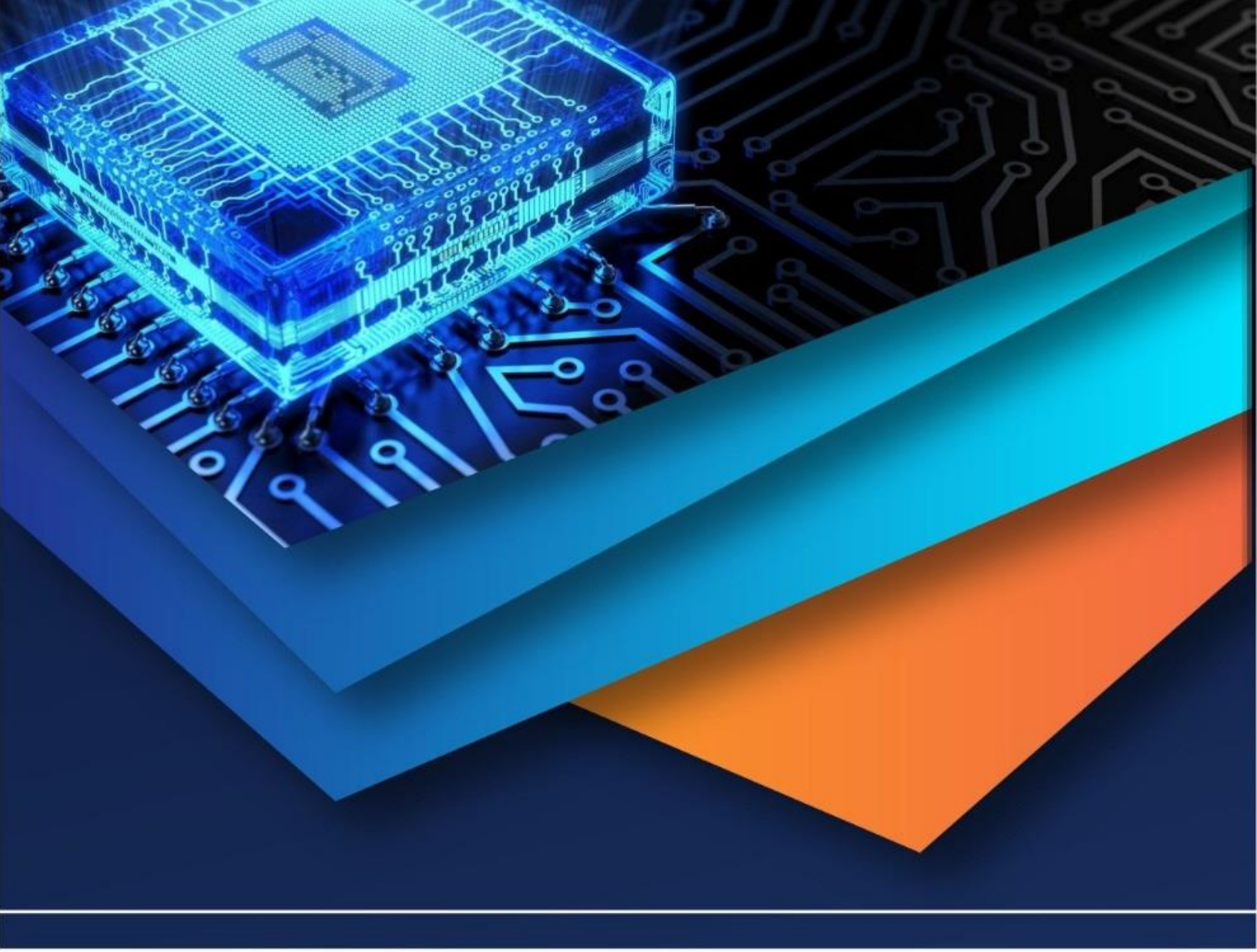



## V. CONCLUSIONS

This paper proposes an efficient secret randomness routing process for transferring data from source node to destination node. Before moving data from the source node to the destination node, we generate a standard secret-key. By using key, the source node and destination node will perform the encryption and decryption process. The source node will enter the transferred message and also take the secret key. Using a private key, the source node will encrypt the communicated message and convert it into a cipher format. After completing the encryption process, the source node will transfer cipher format data to the destination node. The intermediate nodes retrieve cipher format and generate the shortest route randomly. After generating the fastest way, the intermediate node will send cipher format data to the destination node through the shortest path. The destination node will redeem cipher format data and perform the decryption process. By completing the decryption process, the destination node will get original explicit format data. By executing those concepts, we can improve efficiency in the routing process and provide more transferred data security.

## REFERENCES

[1] Mohammad Reza Khalili-Shoja, George Traian Amariucai, Shuangqing Wei and Jing Deng, “Secret Common Randomness From Routing Metadata in Ad Hoc Networks,” IEEE Transactions on Information Forensics and Security ( Volume: 11, Issue: 8, Aug. 2016 ), vol. 11, no. 8, pp. 1674 - 1684, 05 April 2016.
[2] Priyanka Goyal, Vinti Parmar, and Rahul Rishi, "MANET: Vulnerabilities, Challenges, Attacks, Application," IJCEM International Journal of Computational Engineering & Management, Vol. 11, January 2011, vol. 11 January 2011.
[3] Ashish Khisti, Suhas N. Diggavi, and Gregory W. Wornell, "Secret-Key Generation Using Correlated Sources and Channels," IEEE Transactions on Information Theory ( Volume: 58, Issue: 2, Feb. 2012 ), vol. 58, no. 2, pp. 652 - 670, 06 February 2012.
[4] Mukesh Singhal, Rendong Bai, Yun Lin, Yongwei Wang, Mengkun Yang, and Qingyu Zhang, "Key Management Protocols for Wireless Networks," 2012.
[5] S. K. Park and K. W. Miller, “Random number generators: good ones are hard to find,” 2009.
[6] Renato Renner and Stefan Wolf, “Simple and Tight Bounds for Information Reconciliation and Privacy Amplification,” 2005.
[7] Berk Sunnar, “True Random Number Generators for Cryptography,” in Cryptographic Engineering pp 55-73, 2009.
[8] Mohammad Reza Khalili Shoja, George Traian Amariucai, Shuangqing Wei, and Jing Deng, "KERMAN: A Key Establishment Algorithm based on Harvesting Randomness in MANETs," in arXiv:1504.03744 [cs.CR], 2015.
[9] U.M. Maurer, “Secret key agreement by public discussion from common information,” IEEE Transactions on Information Theory ( Volume: 39, Issue: 3, May 1993 ). 39, no. 3, pp. 733 - 742, May 1993.
[10] Qian Wang, Hai Su, Kui Ren, and Kwangjo Kim, "Fast and scalable secret key generation exploiting channel phase randomness in wireless networks," in 2011 Proceedings IEEE INFOCOM, Shanghai, 30 June 2011.
[11] Tim Landstra, Maciej Zawodniok, and S. Jagannathan, "Energy-Efficient Hybrid Key Management Protocol for Wireless Sensor Networks," in 32nd IEEE Conference on Local Computer Networks (LCN 2007), Dublin, 15-18 Oct. 2007.
[12] Mohamed Gaafar, Mohammad Galal Khafagy, Osama Amin, and Mohamed-Slim Alouini, "Improper Gaussian signaling in full-duplex relay channels with residual self-interference," in 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, 22-27 May 2016.
[13] Laurent Eschenauer and Virgil D. Gligor, “A key-management scheme for distributed sensor networks,” in CCS '02: Proceedings of the 9th ACM conference on Computer and communications security, November 2002.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY