Journal of Engineering Science and Technology Review 17 (2) (2024) 197 - 202

Research Article

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

A Trust-Table Propagation Model to Avoid Transmission through Malicious Communication Nodes

Mohammed Al-Momin*

Department of Control and Automation Techniques Engineering, Basra Engineering Technical College (BETC), Southern Technical University (STU), Basra, Iraq

Received 9 December 2023; Accepted 20 April 2024

Abstract

A novel idea for offering a high level of authenticity in data communication has been proposed in this paper. This idea can be coincided by maintaining the transmission of data through highly reputable nodes. A Trust table that measures the authenticity of the end-to-end paths from any node to the various destinations is stored within the communication nodes along with routing tables. Dynamic changes in nodes' authenticities, which are almost due to malicious intrusion, are conceived by periodically propagating discovery packets through the whole network in all directions. Any route that turns unauthentic will be avoided in the upcoming communication operations. When the route is recovered, the trust table will promptly be updated to return this route to used again. The proposed method introduced promising results in dealing with dynamic authenticity changes in nodes. Fast responsive performance was also clearly noticeable in results.

Keywords: Trust Table; Authenticity; Malicious Nodes; Data Communication; Routing.

1. Introduction

Privacy and security are cardinal requirements of efficient data communication. Intruders can exploit nodes, channels, or both to achieve their goals [1-4]. When two end-nodes are communicating, information may need to be transmitted in a top security manner. An eave dropper or what is sometimes termed as hacker is a third-party entity that tries to spy or alter the information travelled through network links or nodes. Researchers interested in the field of data security have taken two directions. Firstly, some researchers worked on encrypting private and secure information when it is being transmitted on the network in order to make information ambiguous to intruders. Encrypting data can add additional computational effort and consequently more time consumption. Many encryption algorithms have been developed for this purpose. The more complex the algorithm used the more secure the data communication [5-6]. Communication protocols need also to be adapted to accommodate such a secure data transmission. For instance, "http" has been modified to its secure version "https" in the seventh layer of OSI model [7-8]. Furthermore, data hiding has also played a major role in this field. Many papers suggested data hiding algorithms and strategies to support information privacy over the network. Authors in [9] proposed an information hiding technique using particle swarm optimization to hide covid 19 data when it is being transmitted among decentralized health centers. Staying in data hiding field, many efforts have been dedicated on embedding critical data into images, audio, or videos, what is well known as steganography [10-13].

However, cryptographic techniques are also subject to attacks. Whenever the cryptographic key is revealed, the whole system fails to offer the desired security and privacy quality. On the other hand, some other papers concentrate on avoiding transmission through suspicious nodes. This method is claimed to be more reliable than transmitting ciphers in public. In fact, it should be taken into account that network nodes and links are also subject to attacks. Even though a particular link or node seems to be authentic at any given time, it can be dominated by intruders later on, and thus it changes to be unauthentic. A node under attack can be utilized to spy on the information passing through it, or may be invested to alter the original data, hence the node is termed to be malicious. Malicious nodes can carry backdoor data or viruses and can consequently break the system's security measures. Malicious nodes can also play a fundamental role in black-hole attack. In black-hole attack, a malicious node illudes other nodes in the network by pretending to offer the shortest path to the destination [14]. In such a kind of attack, the malicious node listens to any route request (RREQ) from its neighboring. As soon as an RREQ is detected, it sends back a route reply message (RREP) pretending to possess a valid route to the destination. In fact, this RREP message is a forged reply since there is no actually assured route to the desired destination, and the main goal behind this operation is to mislead the communicating nodes.

Many research papers have been conducted on detection of black-hole attacks. Paper [15] for instance, utilized two techniques to identify black-hole attack, namely, digital signature and intrusion detection. Authors in [16] suggested a fuzzy neural system with the aid of swarm-based optimization for detecting black-holes. A system investing the capabilities of modified K-Means clustering algorithm and Proportional Coinciding Score (PCS) has been suggested in [17] aiming to detect black and sink holes. Paper [18] on the other hand, detected malicious nodes using time, special, and event correlations. Many researchers concentrated on developing a system to identify malicious nodes in wireless ad hoc and sensor networks using machine learning strategy [19-22]. Clone attack is another challenge that faces secure transmission in computer Wireless Sensor Networks (WSN). In clone attacks, termed as node replication attack as well,

*E-mail address: mohammed.al-momin@stu.edu.iq
ISSN: 1791-2377
doi:10.25103/jestr.172.21

attacker makes several replicas of the infected node, and distribute these nodes in different locations of the network. All the created copies of the malicious node have the same valid credentials and legitimacy as the compromised the original node. Consequently, other nodes deal with these infected nodes as authentic ones [23-25]. In [26], researchers proposed a Hybrid Clone Node Detection strategy to detect redundant nodes that likely to be clones. In [27], a cryptographic that benefits from both AES and ECC cryptography methods in order to tackle clone attacks.

One of the major drawbacks of the previously mentioned schemes is their inability to deal with network dynamics. In other words, a node may be infected or attacked at any time. A link as well may be turned to be insecure at any arbitrary time. Even though a link or node has been authenticated in advance, the authentication certification needs to be renewed and updated from time to time. In this paper, a new approach, that efficiently deals with sudden changes in the network authenticity changes, has been suggested. This method reads nodes and links authenticities periodically through the propagation of a trust table, which is filled and updated according to nodes reputation index. Any degradation in the authenticity of a particular node will be sensed at the first blush by the neighboring nodes, then the experience of these nodes will be propagated through other network entities.

2. Method

Malicious nodes are considered a serious challenge in communication networks. Consequently, efforts need to be devoted to addressing this problem. Black-hole attack is one of the major forms of such a challenge. Figure 1 below explains the theory behind this kind of attack. Suppose that Node 1 is the source node, and Node 4 is the destination one. The valid path from the source to destination passes through Node 3. Whenever Node 1 wants to communicate with Node 4, it propagates a Route Request (RREQ) packet to all the neighboring nodes. Since Node 3 is not targeted, it propagates the request to other connected nodes aiming to detect a valid path to the destination. When destination is reached, a Route Reply (RREP) is sent back to the source node through the same nodes used for transmitting the request, but in a reverse order. Node 2, as a malicious node, is always in a waiting state, where it sends back an RREP to Node 1 pretending to offer the best route to the destination node even though there is no real path to the required destination. Whenever the earliest RREP packet reaches the source node, it ignores all the subsequent responses supposing that the node which replied first is offering the shortest path to the desired destination. When a fake connection is established, the malicious node either drops the communication packets or forwards them to an eve-dropping third party.

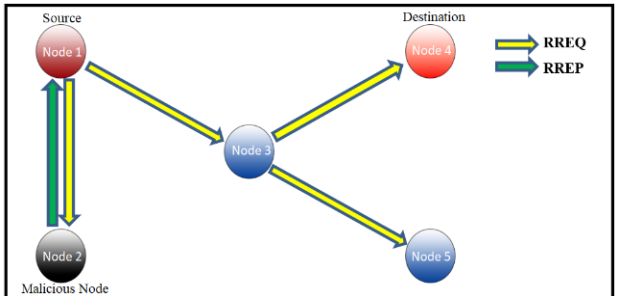


Figure 1. The problem of black-hole attack

OMMet++ was used in this work to suggest a solution to malicious nodes attacks by adopting a reputation-based routing technique. A network of 12 nodes, shown in Figure 2, was used to examine the performance of the suggested model. This model adopts the idea of sharing the experience of nodes regarding the authenticity of each other. This experience is propagated through the network as a trust table. Each node has its own trust table. A trust table entry measures the authenticity of a particular neighboring node in communicating with a targeted destination. A low trust index may not necessarily mean that this node is malicious or attacked, but it further means that a certain node in the whole path to the destination is of bad reputation. In run time, as soon as a node misses its good reputation, the first nodes to sense such a degradation in its authenticity are neighbors. In consequence, these neighbors will inform their own other neighbors of such a change in authenticity in order to update their trust tables. The nodes in figure 2 represent routers, therefore they possess routing tables. Our proposed scheme supposes that a trust table should be included within these routers as well to assess the authenticities of different alternative paths to the destination.

In the beginning, the different network's nodes propagate request messages in all directions. When the destination node receives a request, it sends back a reply to the source node with the path's quality parameters such as path's length and security. As a result, the source node receives several replies from the destination node corresponding to different alternative paths. Finally, the source node uses the paths' information of the received replies to set up their own routing and trust tables. In order to keep track of any changes that happen to the network in term of authenticity at any arbitrary time, the source node continuously sends a discovery packet in a random direction from time to time to investigate possible changes in path preferences. In this work, the network is trained in the beginning to establish nodes' routing and trust tables then Node 2 is assigned the role of destination in order to dimension demands on different network's links leading to the designated destination. A network disorder has inadvertently been inserted by attacking Node 5 at a predefined time to examine the behavior of the proposed system against any security issue caused by a malicious node.

The following algorithm clearly explains the overall methodology used in the proposed scheme in a time order manner.

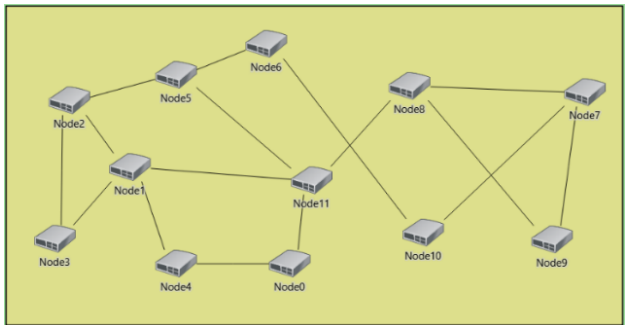


Fig. 2. The Simulated Network

Step 1- Initialize the routing and trust tables of the different nodes: A routing table entry contains the distance to a specific destination through a particular output gate. Similarly, the trust table entry contains the trust index of this output gate when communicating with the specified destination. Therefore, a communication node can have a number of entries in its trust table to a specific destination each is

associated with a particular output gate. These entries may have different values of trust indices. A typical trust-table entry is shown in Figure 3. Distances and trust indices to all destinations are initialized to infinity.

Fig. 3. A Typical Trust-Table Entry

Step 2- Generating training packets: A total of 500 packets are generated from the different nodes and forwarded in all directions targeting the various destination nodes. The destination node in turn responds by sending back a reply to the same source of the reached requesting packet. The followed path is recorded in a stack in for two reasons, firstly, to follow the same path when a reply packet is returned, and secondly, to prevent the requesting packet from visiting a node twice.

Step 3- Updating the routing and trust tables: Whenever a reply packet is returned from the destination, the source node updates its routing and trust tables according to the data stored in the packet. This data includes the followed path's distance, and the experienced trust quality. If the path's length of the reply packet is shorter than the value already stored in the routing table entry, the distance field in the entry is replaced by the new length, otherwise no update is required. The trust table on the other hand, is updated as long as the followed path experiences better security than the default path.

Step 4- After a network has been trained, communication between source and destination nodes will take place on the path which had been set up during training phase. Although all the future communication between these source and destination nodes will follow this route which has been proved to be optimal, discovery request packets are sent periodically in random directions to sense any change in security and connectivity that may occur in the network.

Step 5- A particular output gate will be used by a node to forward the current request packet in a probability of P, which is calculated according to eq (1).

$$P = \frac{\omega}{\omega_T} * 100\% \quad (1)$$

Where ω is the gate's weight, and ω_T is the total sum of weights of all the node's output gates. This equation stimulates the packet to follow the optimal source-destination path. This path is characterized by possessing the maximum weight. The weight is calculated to compromise both path's length and authenticity. Where a path of a greater weight is described to have a smaller number of trusted hops. This optimal path is used to establish the default connection between the source and destination nodes for forthcoming communications.

Step 6- The current message could be forwarded through a non-default gate in a probability of Pd which is described in eq (2). When a request follows a non-default connection, it is termed as a discovery request. This request is replied whenever it reaches the destination node with a discovery reply packet. If the discovery reply packet offers a better quality of service than the default route, the trust and routing tables are updated accordingly to capture this path. However, output gates' weights are tuned to give a greater bias toward the default optimal route in order not to consume the network communication time in excessive discovery operations.

$$Pd = 1 - P = (1 - \frac{\omega}{\omega_T}) * 100\% \quad (2)$$

This strategy of propagating discovery requests repetitively proved to be a very efficient means of keeping the routing system aware of any trust or connectivity issue that may occur at any time.

Step 7- To test the system's performance in the occasion of any authenticity issue, node's 5 authenticity has been tarnished deliberately in an arbitrary time. Simulation showed the substantial influence of node's reputation among other nodes. This was associated with the decreased number of requests passing through this disrepute node.

Step 8- Security issue of node 5 has then been resolved to assess the capability of the proposed routing system to reconsider recovered nodes.

The complete algorithm for updating trust and route tables is represented by an explaining flow chart in Figure 4.

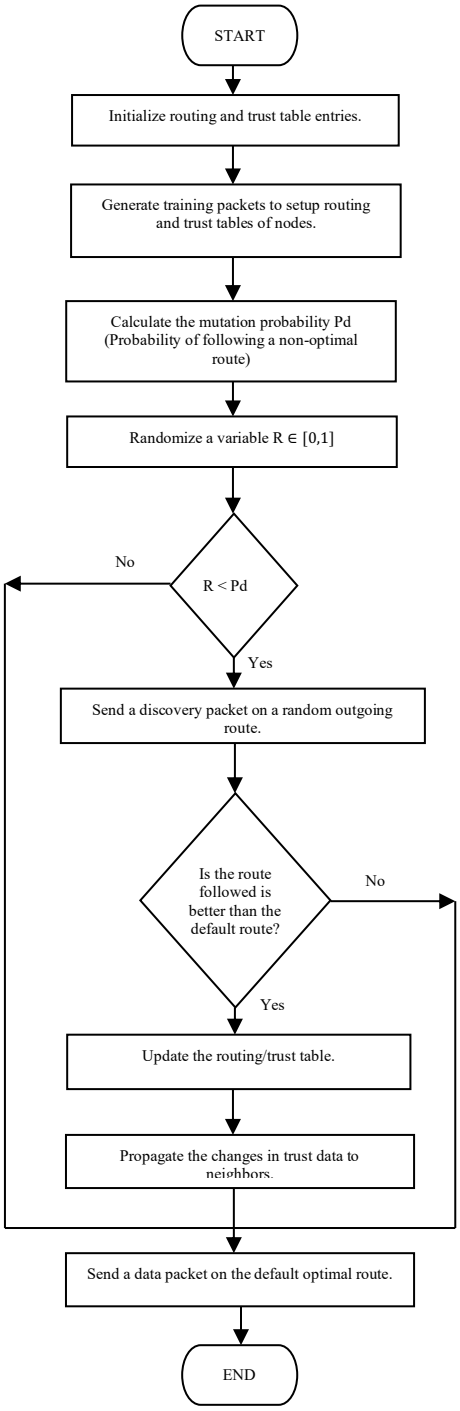
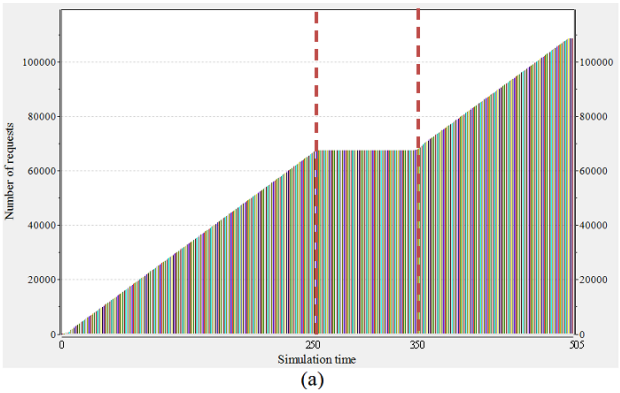


Fig. 4. Updating trust and route tables process.

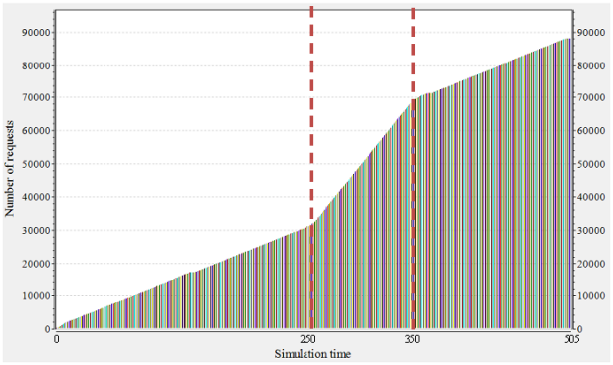
3. Results and Discussion

Simulating the network using omnet++ gave promising results in terms of the capability of adaptivity when any potential attack is subjected to the system. Node 7 was considered a destination node in order to test the performance of the proposed routing scheme. Figure 5 shows the accumulative number of requests influence on nodes 8, 6 and 11 respectively in case of node 8 being attacked. Node 8 is supposed to be compromised temporarily during the time interval 250 -350. During this period, node 8 had missed its reputation among other nodes. This degrade in trust is first sensed by the neighboring nodes, then other nodes start to update their outgoing routes to the destination consecutively. This interprets the stop of any further requests during this period shown in Figure 5(a). Node 6, on the other hand, had been greatly affected by this event since it became the only bridge to the destination, node 7. Therefore, requests on node 6 had dramatically increased during this time, as clearly shown in Figure 5(b). Figure 5(c) shows that node 8 has also been affected, but to a minor extent, when node 8 is attacked. This is because node 11 is not only playing a role in transmitting data to the destination though the malicious node, but it also serves to connect with node 6 and other neighboring nodes. For this reason, the number of requests on this node has decreased but to a mirror extent.

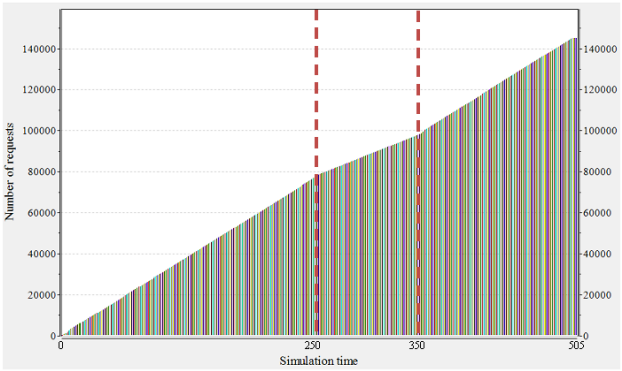
Figure 6 shows the accumulative number of requests on nodes 10, 6 and 8 respectively on an occasion where node 10 is being compromised. Again, Figure 6(a) shows an instant response by neighbors to avoid transmitting through node 10. Node 6, on the other hand, lost its fame in communicating with the destination. As a result, requests on it had substantially decreased leaving only the requests generated by node 6 itself as depicted in Figure 6(b). Figure 6(c) shows the accumulative number of requests on node 8 when node 10 is attacked. It is clear from this figure that there is a minor increase in the level of congestion on node 8 due to compromising node 10. This is because node 8 has already been offering the best path to the destination for most of the connected source nodes even when node 10 has not been yet attacked. Figure 7 clarifies the accumulative number of requests on node 6 when node 11 is attacked which experienced a dramatical increase since node 6 is now representing the only trusted gateway to the destination for most nodes. On the other hand, compromising node 6 will not significantly affect congestion on node 11 as shown in Figure 8. This is because node 6 was not representing the optimal gateway to the destination for most nodes from the beginning and even when node 6 was not attacked.



(a)

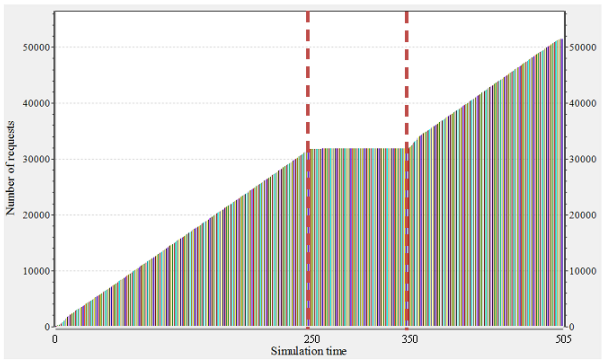


(b)

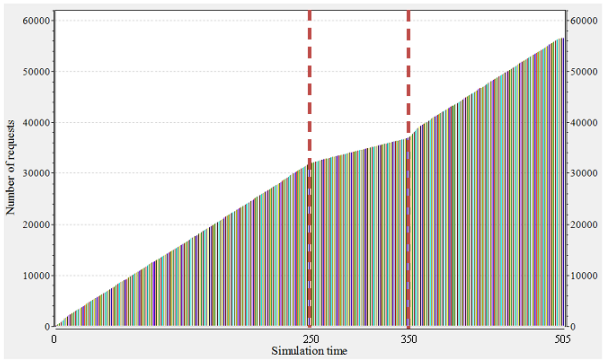


(c)

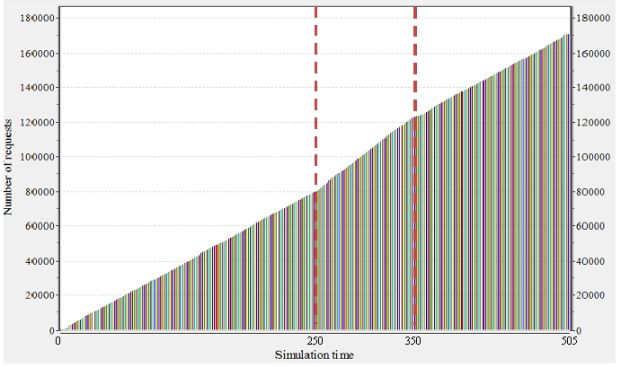
Fig. 5. (a) Accumulative requests on node 8 when node 8 is attacked, (b) Accumulative requests on node 6 when node 8 is attacked and (c) Accumulative requests on node 11 when node 8 is attacked



(a)



(b)



(c)

Fig. 6. (a) Accumulative requests on node 10 when node 10 is attacked, (b) Accumulative requests on node 6 when node 10 is attacked and (c) Accumulative requests on node 8 when node 10 is attacked

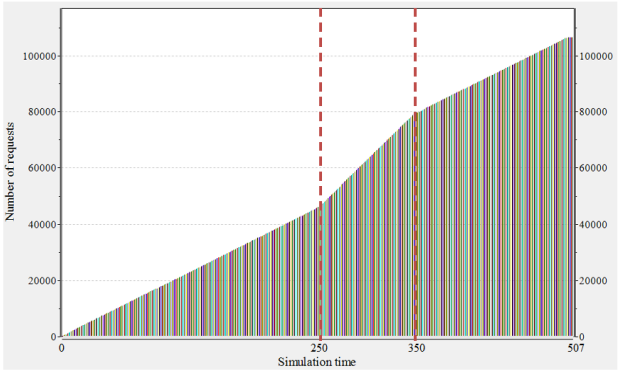


Fig. 7. Accumulative requests on node 6 when node 11 is attacked.

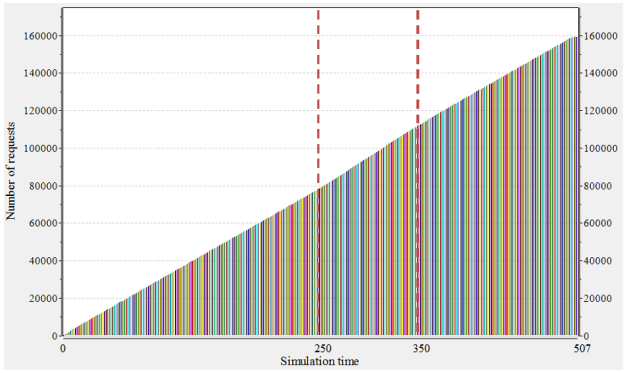


Fig. 8. Accumulative requests on node 11 when node 6 is attacked.

A comparison with the conventional already-existing trust-independent routing strategies can be established by switching-off the trust-table capability in the simulated model. In order to accentuate the difference, requests on node 6 have been studied when node 8 is considered malicious. Figure 9(a) shows the accumulative requests on node 6 when trust-aware routing strategy is adopted, whereas Figure 9(b) shows these requests when this facility is disabled. It is pretty clear that the proposed model offers a more adaptive solution to prevent transmission through malicious nodes in the presence of any network inconvenience. One Should keep in mind that the increased number of requests on node 6 in the proposed scheme is attributed the fact that this node constructs the best alternative secure route to the destination when node 8 is attacked.

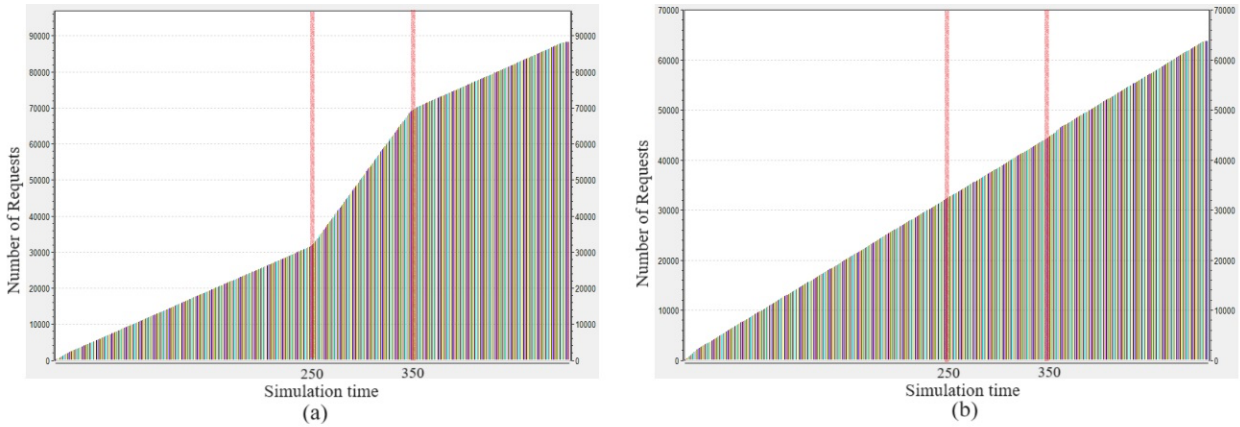


Fig. 9. Accumulative requests on node 6 when node 8 is attacked. a)With trust-aware routing facility and b)Without trust-aware routing facility

4. Conclusions

Trust-aware routing may be considered as one of the most promising adaptive routing strategies. In this paper a new routing strategy has been considered, that takes into account the dynamic changes in nodes' trustworthiness when forwarding data to the designated destination. This has been achieved by creating a trust table that reflects the amount of authenticity of the different intermediate communication nodes when communicating with a specific destination. Changes in node's authenticity are first sensed by its neighbors and then propagated to the other nodes. In consequence, the trust tables and hence routing tables need to be updated periodically during run time to recognize any change in nodes' security levels. The proposed algorithm has been tested for different scenarios with different nodes had been attacked aiming to examine the performance of this new routing scheme. The results were promising in terms of the system ability to update the wights of the communicating paths in occasion when any security inconvenience occurred, and the fast response to dynamic changes in security coordinates. For future work, mobility of malicious nodes in ad-hoc networks can also be considered and recognized to prevent transmission through these nodes without a real necessity to re-propagate an updated version of trust tables when these nodes change their locations.

References

[1] J. Chen, J. Zhang, Z. Chen, M. Du, and Q. Xuan, "Time-Aware Gradient Attack on Dynamic Network Link Prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 2091-2102, Feb. 2023, doi: 10.1109/TKDE.2021.3110580.

[2] M. Conti, N. Dragoni, and V. Lesyk, "A Survey of Man in the Middle Attacks," *IEEE Communic. Surv. Tutor.*, vol. 18, no. 3, pp. 2027-2051, Mar. 2016. doi: 10.1109/COMST.2016.2548426.

[3] R. Moura, D. R. Matos, M. L. Pardal, and M. Correia, "MultiTLS: Secure Communication Channels with Cipher Suite Diversity," in *ICT Sys. Secur. Priv. Protect.* Hölbl, M., Rannenberg, K., Welzer, T. Ed., SEC 2020. IFIP Advances in Information and Communication Technology, vol 580. Springer, Cham. https://doi.org/10.1007/978-3-030-58201-2_5.

[4] M. M. S. A. Al-Momin, J. Cosmas, and S. Amin, "Adaptive three-layer weighted links routing protocol for secure transmission over optical networks," *WSEAS Transact. Communic.*, vol. 11, no. 8, pp. 287-298, Aug. 2012.

[5] H. Abroshan, "A Hybrid Encryption Solution to Improve Cloud Computing Security using Symmetric and Asymmetric Cryptography Algorithms," *Int. J. Adv. Comp. Sci. Applicat.*, vol. 12, no. 6, pp. 31-37, Dec. 2021, doi: 10.14569/IJACSA.2021.0120604.

[6] M. Jammula, V. M. Vakamulla, and S. K. Kondoju, "Performance Evaluation of Lightweight Cryptographic Algorithms for Heterogeneous IoT Environment," *J. Interconn. Netwok.*, vol. 22, pp. 2141031:1-2141031:21, Jan. 2022, doi: 10.1142/S0219265921410310.

[7] Q. Hu, M. R. Asghar, and N. Brownlee, "A large-scale analysis of HTTPS deployments: Challenges, solutions, and recommendations," *J. Comput. Secur.*, vol. 29, no. 1, pp. 25-50, Feb. 2021, doi: 10.3233/JCS-200070.

[8] S. Vandeven, Aug. 2019, "Information Security Reading Room SSL / TLS : What ' s Under the Hood," *SANS Institute Reading Room*, [Online]. Available: https://sansorg.egnyte.com/dl/l3PuV0zPMK

[9] A. H. Mohsin *et al.*, "PSO–Blockchain-based image steganography: towards a new method to secure updating and sharing COVID-19 data in decentralised hospitals intelligence architecture," *Multimed. Tools Appl.*, vol. 80, no. 9, pp. 14137–14161, Apr. 2021, doi: 10.1007/s11042-020-10284-y.

[10] M. Al-Momin, I. A. Abed, and H. A. Leftah, "A new approach for enhancing LSB steganography using bidirectional coding scheme," *Int. J. Electr. Comp. Engin.*, vol. 9, no. 6, pp. 5286-5294, Dec. 2019. doi: 10.11591/ijece.v9i6.

[11] X. Duan, D. Guo, N. Liu, B. Li, M. Gou, and C. Qin, "A New High Capacity Image Steganography Method Combined with Image Elliptic Curve Cryptography and Deep Neural Network," *IEEE Access*, vol. 8, pp. 25777-25788, 2020, doi: 10.1109/ACCESS.2020.2971528.

[12] P. Rakshit, S. Ganguly, S. Pal, A. A. Aly, and D. N. Le, "Securing technique using pattern-based LSB audio steganography and intensity-based visual cryptography," *Comp., Mater. Contin.* vol. 67, no. 1, pp. 1207-1224, Jan. 2021, doi: 10.32604/cmc.2021.014293.

[13] M. Fuad and F. Ernawan, "Video steganography based on DCT psychovisual and object motion," *Bullet. Electr. Engin. Inform.*, vol. 9, no. 3, pp. 1015-1023, Jun. 2020, doi: 10.11591/eei.v9i3.1859.

[14] P. Krishnan, and P. Kumar, "Detection and Mitigation of Smart Blackhole and Gray Hole Attacks in VANET Using Dynamic Time Warping," *Wireless Pers. Commun.,* vol. 124, pp. 931–966, Dec. 2021, doi: https://doi.org/10.1007/s11277-021-09390-3

[15] M. I. Talukdar, R. Hassan, M. S. Hossen, K. Ahmad, F. Qamar, and A. S. Ahmed, "Performance Improvements of AODV by Black Hole Attack Detection Using IDS and Digital Signature," *Wirel. Commun. Mob. Comput.*, vol. 2021, pp. 1 – 13, Mar. 2021, doi: 10.1155/2021/6693316.

[16] H. Moudni, M. Er-Rouidi, H. Mouncif, and B. El Hadadi, "Black Hole attack Detection using Fuzzy based Intrusion Detection Systems in MANET," in *Proc. Comp. Sci.*, Netherlands, Elsevier B.V., vol. 151, 2019, pp. 1176-1181, doi: 10.1016/j.procs.2019.04.168.

[17] R. K. Dhanaraj, L. Krishnasamy, O. Geman, and D. R. Izdrui, "Black hole and sink hole attack detection in wireless body area networks," *Comput., Mater. Contin.*, vol. 68, no. 2, 2021, doi: 10.32604/cmc.2021.015363. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050919306350

[18] Y. Lai et al., "Identifying malicious nodes in wireless sensor networks based on correlation detection," *Comput. Secur.*, vol. 113, Feb. 2022, doi: 10.1016/j.cose.2021.102540.

[19] A. J. Clement Sunder and A. Shanmugam, "Black Hole Attack Detection in Healthcare Wireless Sensor Networks Using Independent Component Analysis Machine Learning Technique," *Curr. Signal Transduct. Ther.*, vol. 15, no. 1, pp. 56-64, Jul. 2018, doi: 10.2174/1574362413666180705123733.

[20] M. Abdan and S. A. H. Seno, "Machine Learning Methods for Intrusive Detection of Wormhole Attack in Mobile Ad Hoc Network (MANET)," *Wirel. Commun. Mob. Comput.*, vol. 2022, Jan. 2022, doi: 10.1155/2022/2375702.

[21] M. C. Belavagi and B. Muniyal, "Multiple intrusion detection in RPL based networks," *Int. J. Electr. Comp. Eng.*, vol. 10, no. 1, pp. 467-476, Feb. 2020, doi: 10.11591/ijece.v10i1.pp467-476

[22] S. Laqtib, K. El Yassini, and M. L. Hasnaoui, "A technical review and comparative analysis of machine learning techniques for intrusion detection systems in MANET," *Int. J. Electr. Comp. Eng.*, vol. 10, no. 3, pp. 2701-2709, Jun. 2020. doi: 10.11591/ijece.v10i3.pp2701-2709.

[23] M. Keerthika and D. Shanmugapriya, "Wireless Sensor Networks: Active and Passive attacks - Vulnerabilities and Countermeasures," *Global Trans. Proceed.*, vol. 2, no. 2, pp. 362-367, Nov. 2021, doi: 10.1016/j.gltp.2021.08.045.

[24] H. R. Shaukat, F. Hashim, M. A. Shaukat, and K. A. Alezabi, "Hybrid multi-level detection and mitigation of clone attacks in mobile wireless sensor network (MWSN)," *Sensors (Switzerland)*, vol. 20, no. 8, Apr. 2020, doi: 10.3390/s20082283.

[25] J. R. Dora and K. Nemoga, "Clone Node Detection Attacks and Mitigation Mechanisms in Static Wireless Sensor Networks," *J. Cybersecurity Priv.*, vol. 1, no. 4, pp. 553-579, Sep. 2021, doi: 10.3390/jcp1040028.

[26] P. P. Devi and B. Jaison, "Protection on Wireless Sensor Network from Clone Attack using the SDN-Enabled Hybrid Clone Node Detection Mechanisms," *Comput. Commun.*, vol. 152, pp. 316–322, Feb. 2020, doi: 10.1016/j.comcom.2020.01.064.

[27] V. Mohindru, Y. Singh, and R. Bhatt, "Hybrid Cryptography Algorithm for Securing Wireless Sensor Networks from Node Clone Attack," *Rec. Adv. Elec. Electr. Eng. (Formerly Rec. Pat. Elec. Electr. Eng.)*, vol. 13, no. 2, pp. 251-259, Apr. 2020, doi: 10.2174/2352096512666190215125026.