International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 06 | June - 2022 Impact Factor: 7.185 ISSN: 2582-3930

BLOCKCHAIN BASED SECURE ANDENERGY EFFICIENT ROUTING PROTOCOL FOR WSN

Guided by Mr. K. Karthik, Assistant Professor / ECE, VCET

Abid Mohammed Anusudeen, Srinath J, Viswanathan P

Electronics and Communication Engineering, Velammal College of Engineering and Technology, Madurai

ABSTRACT Routing in a Wireless Network Sensor (WSN) is a critical process as it is basically responsible for transmitting the data to Base Stations (BS). WSNs are used in a wide range of applications that require the data to be transmitted and received securely, such as in battlefield surveillance, flood detection, forest fire detection, traffic surveillance, etc. But the actual routing process is susceptible to attacks that can completely destroy the operation of the WSN. The intervention from malicious nodes is one of them. Malicious nodes can send erroneous information of particular lengths to other nodes in order to increase their chance of receiving the data that is being transmitted. The malicious nodes can also send corrupted data to other nodes and cause the operation of WSNs to fail. Hence, a trustworthy routing system is a must in order to ensure that the WSN will operate smoothly and efficiently.

An authentication scheme is introduced that is blockchain-based, so as to provide secure routing in the WSNs. As there are unauthenticated and malicious nodes that affect the routing process, the correct identification of the routing path becomes a challenging issue. Therefore, in our model, to prevent the involvement of these malicious nodes in the network, the registration of the nodes is done by a Certificate Authority Node. Each node that is involved in the routing gets authenticated by the BS and a mutual authentication is also performed. Furthermore, in the proposed routing protocol, a Cluster Head (CH)is used. The selection of a CH node is based on the residual energy of the node and the node which has the minimum distance from BS. This CH collects the data from the other nodes and sends it forwards to the BS. Every node can authenticate the process of transmitting or receiving data using a Proof of Authority (PoA) method in the blockchain network.

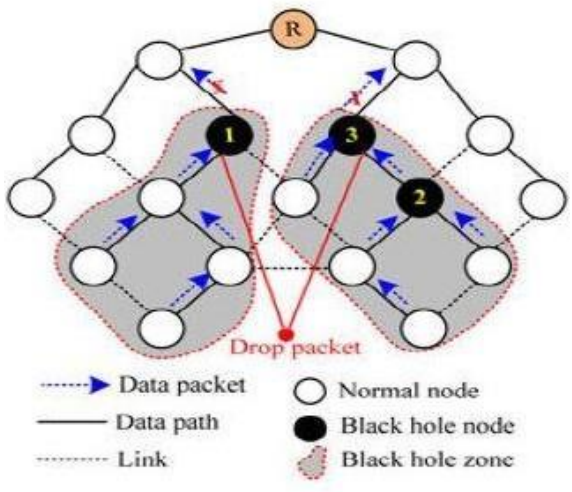
INDEX TERMS Wireless sensor networks, trusted routing, bidirectional blockchain, deep learning, Markov decision.

1. INTRODUCTION

The multi-hop routing technique is a key feature of WSN technology. Nonetheless, multi-hop routing is vulnerable to a number of attack types due to the distributed and dynamic nature of WSN, weakening security. In order to maximize the possibility of receiving packets, a malicious node may release erroneous queue length information,

causing other routing nodes' routing schedules to be altered. Because it is difficult to discern between two routing nodes' real-time changes in routing information, current routing algorithms have difficulty identifying such rogue nodes.

When a malicious node receives data packets from a neighbour node, instead of forwarding them to the next-hop neighbour node, it discards them right away. This creates a data "black hole" in the network, which might be difficult to spot in WSNs for routing nodes, shown in the below figure. External attackers or legitimate internal nodes intercepted by external attackers could be the source of these fraudulent nodes.



In recent years, blockchain technology and routing algorithms have been the subject of extensive research.The blockchain is a decentralized network that is maintained by multiple nodes and is primarily concerned with trust and security issues.

The security mechanism based on cryptography and identity verification is not appropriate to deal with improper behavior attacks of nodes. Because the premise of implementing these security mechanisms is that all nodes are cooperative and trustworthy which is unrealistic for internal attacks on the network. Also, these mechanisms also require complex calculations and high memory capacity, which additionally leads to high energy consumption. Therefore, the trust perception based security mechanisms have been proposed to solve the problems in the security mechanisms based on encryption and identity verification.

To address this, Bidirectional Blockchain is employed, which combines the advantages of multiparty computation (MPC) with blockchain to assure the anonymity of keys used to sign distributed data, as well as the correctness and integrity of the data received. The forward blockchain contains the data signed by its owner, as well as the hash digest of the previous block (as in a normal blockchain), and the reverse blockchain contains the public keys used to verify the authenticity of the data stored in the forward blockchain, as well as the hash digest of the subsequent block (as in a normal blockchain).

The rest of this article is as follows: Along with some preliminary information, Section 2 summarizes current strategies for a reliable routing method in WSNs. Section 3 discusses the suggested trustworthy routing model. Section 4 presents many experimental results that demonstrate the suggested model's efficacy. Finally, in Section 5, we will complete the paper and outline future goals.

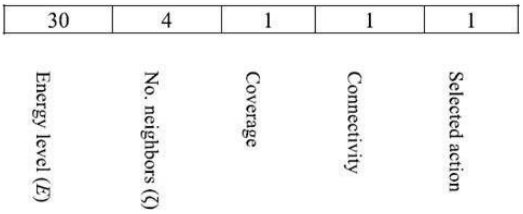
2. EXISTING SYSTEM

The existing system offered a trusted routing method that combines deep blockchain and Markov Decision Processes (MDPs) in order to enhance the routing security and efficiency of WSNs. To authenticate the process of transmitting the node, the proposed approach utilizes a Proof of Authority (PoA) method inside the blockchain network. The validation group necessary for proving is selected using a deep learning methodology that focuses on the properties of each node. MDPs are then used to choose the appropriate next hop as a forwarding node capable of transferring messages simply and securely.

The main goal of this proposed method is to create a secure routing system for wireless sensor networks by combining deep chain and Markov decision-making. The fundamental architecture of the proposed system is shown, and it consists of three phases: creating a node data structure, picking a validator using a deep learning model, and optimizing the next hop using MDP. The next subsections go through each of these steps in detail.

SĒep 1: Build Node DaĒa SĒrucĒure

At start, all sensors are identical and serve no purpose as validators or slave nodes. They are not anonymous sensors; each one has a unique ID (e.g., anonymous addresses). A transmission's packets are all the same size. In a wireless sensor network, there are two forms of data transmission: direct transmission and multi-hop data transfer. In this case, multi-hop data transport is used. Each cell in the WSN starts with the same amount of energy and remains static with symmetrical communication. The function of any node that was previously set to unstated is turned into the validator or minion upon initialization.Each node in the network keeps a data structure that contains a variety of node properties, such as the chosen action (validator or not), the energy level, the coverage, the interconnectivity, and the number of its neighbours.



SĒep 2: ValidaĒors ElecĒion UsingDeep Neural Learning

After establishing the data structure for each node, the characteristics of these nodes are utilized to determine the most significant nodes that will act as validators in the blockchain proof framework's authentication network. A deep neural network is used to make the selection. Deep learning techniques are used to learn functional hierarchies, in which the features are constructed on higher levels using minor levels. The activation potentials supplied by each of the first hidden layer's unique input measurements are utilized to choose the most appropriate functions. The features are selected to provide more accurate classifications than the high-dimensional initial characteristics. The stacked RBMS (Deep Belief Network) is used as a BlackBox with its default settings.

SĒep 3: Blocfichain Based RouĒing NeĒworfis

To improve the credibility and robustness of routing information, blockchain technology

is incorporated into the wireless sensor network and blockchain token transactions are used to trace node-related information. Blockchain is a distributed database with some tamper-proof, decentralization, and information traceability characteristics. As it is much more effective at processing transactions, the PoA consensus method is employed for blockchain networks.

SĒep 4: NexĒ Hop SelecĒion Using MDPS

In finite horizon problems, MDP is used to find the best strategy for maximizing a value function, which is defined as the expected sum of rewards across all decision epochs, or as the anticipated total discounted reward or the expected average reward in infinite horizon problems.

3. TRUST MODEL BASED SECURE ROUTING

Calculating trust levels helps the nodes discover malicious nodes. The key to securing routing is figuring out how to raise the trust value of legitimate nodes while rapidly lowering the trust value of malicious nodes. As a result, the adaptive penalty coefficient is utilized to rapidly diminish the trust value of malicious nodes, achieving the goal of immediately identifying and deleting malicious nodes.

DirecĒ TrusĒ Value

To build a trust relationship between nodes, the node's behaviour must be transformed into a value that indicates the degree of trust, which can be written as,

$$DT_{ij}^{t}=\gamma * HT_{ij}^{t}+ (1-\gamma)*(R_j+S_j)^t$$

where $\overset{t}{HT}_{ij}$ represents the historical trust value of node I after volatilization as evaluated by node j.

Therefore, in order to accelerate the reduction of the trust value of node j, the volatilization factor λ is introduced to reduce the effect of the historical trust value. The expression formula of historical trust value is given as

$$HT_{ij}^{t}=\lambda(DT_{ij}^{t-1}+HT_{ij}^{t-1})$$

where λ controls the influence of the historical trust value on the current direct trust value.

$$R_j=\frac{\theta*receive\_message_j\text{-}rejection_j}{message_j}$$

$$S_j=\frac{\theta*send\_message_j\text{-}un\_send_j}{message_j}$$

where $t(R_j+S_j)$ represents the trust value in the current state, γ and (1 - γ) respectively represent the weight of the trust value after the historical trust value volatilized and the trust value in the current state. $0 < \gamma < 1$, and the value depends on the specific WSNs. $reveive\_message_j$ indicates that node i monitors the number of data packets received by node j, and $send\_message_j$ indicates that node i monitors the number of data packets sent by node j. message j represents the total number of data packets received and sent by the monitored node j. $rejection_j$ and un_send_j respectively represent the number of data packets that node j refuses to receive and refuse to send. Considering the importance of quickly identifying malicious nodes, we define an adaptive penalty coefficient θ expressed as

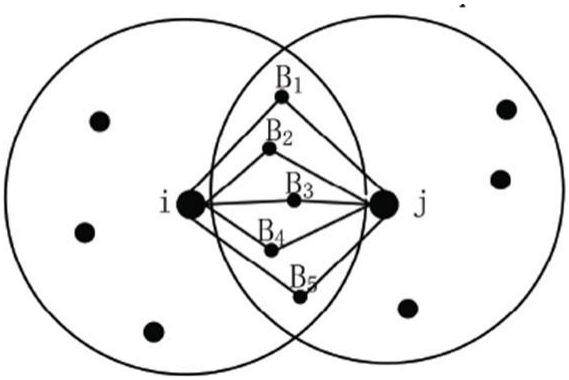
where, q is the nu$\theta = \frac{-a_1}{1+e^{-a_2 * AC_j + a_3}} + 1$stedneighbor nodes, DT_{uj}^t is the direct trust value that node i evalua $AC_j = \frac{AB_j}{NB_j}$ DT_{uj}^t represents where AB_j and N_j the direct trust value that node u evaluates node j.

the abnormal behavior and normal behavior of node j. a_1, a_2 and a_3 are the adjustable parameters of the adaptive penaltycoefficient.

IndirecĒ TrusĒ Value

It needs a significant quantity of communication energy and may cause data congestion. This is because node i must first ask the public trusted neighbour node u for the direct trust value of node j before calculating the indirect trust value.

As shown in the figure below, the public trusted neighbor node u is a member of $B_h = B_h [B_1, B_2, B_3, \ldots, B_q], u \in B_h$



Therefore, in order to avoid the transmission of a large amount of query information between nodes, this paper adopts a centralized computing mode to reduce the burden on nodes.

The calculation formula of indirect trust value that node i evaluates node j is expressed as follows:

$$IT_{ij}^t = \frac{1}{q} \sum_{u \in B_h}^{q} (DT_{iu}^t * DT_{uj}^t)$$

Energy TrusĒ Value

In the network, there may be a situation where the trust value of a node is high but its remaining energy is low, which makes the node die prematurely, thereby affecting the structure and energy consumption of the entire network.

$$E\_receive_j = l * E_{elec}$$

$$E\_send_j = \begin{cases} l * E_{elec} + l * \varepsilon_{fs} * d^2 & d < d_0 \\ l * E_{elec} + l * \varepsilon_{mp} * d^4 & d \geq d_0 \end{cases}$$

$$d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{mp}}$$

where Eelec is the radio frequency energy consumption coefficient of the nodes, and l is the size of messages and data packets. The initial energy of node j is expressed by 0 E and the remaining energy is expressed by RE_j which is shown as

$$RE_j = E_0 - E\_receive_j - E\_send_j$$

Therefore, the energy trust value of node jis:

$$E_j = \frac{RE_j}{E_0}$$

Comprehensive TrusĒ Value

Three factors make up the total trust value: direct trust value, indirect trust value, and energy trust value. It represents the nodes' level of trust. The higher the trust level, the higher the complete trust value of the nodes. If node I determines that node j's

comprehensive trust value is less than the CTth threshold, node I considers node j to be malicious and removes it from the network, preventing it from engaging in any network operations. As a result, the node I to node j comprehensive trust value is stated asfollows:

$$CT_{ij}^{t} = \eta_1 * DT_{ij}^{t} + \eta_2 * IT_{ij}^{t} + \eta_3 * E_j$$

where, η_1 , η_2 and η_3 are the weights of direct trust value, indirect trust value and energy trust value, respectively, $\eta_1 + \eta_2 + \eta_3 = 1$.

ClusĒer Head SelecĒion

First, each node broadcasts the ID of the neighbor with the largest comprehensive trust value. Secondly, after receiving the packets, the neighbor node checks whether the packet matches its ID. If it matches, the number of times it is elected will be

increased by one (Elected_num + 1). Finally, each node broadcasts a packet with Elected_num, and the node with the highest Elected_num serves as CH.

TrusĒ Value UpdaĒe

The calculation and update of the trust value are the foundation and focus of the trust model. Specific updating steps are described as follows.

SĒep 1: MNs monitor the normal and abnormal behaviors of the neighbor nodes, and use formula (1) to evaluate the direct trust values of the neighbor nodes.

SĒep 2: After CHs, INs and routes are determined, the network begins to enter the stable communication stage like LEACH.

SĒep 3: when entering the last time slot of the stable phase, MNs attach the direct trust values of the evaluated neighbor nodes and their remaining energy to the data packet.

SĒep 4: MNs send packets to their CHs, and then CHs send them to the Sink in a multi-hop manner. Finally, the Sink calculates the indirect trust value and the comprehensive trust value according to the direct trust value.

SĒep 5: The Sink sends the calculated comprehensive trust value to each CH by multicast, and CHs forward it to MNs after receiving it. So that the node can update the comprehensive trust value for neighbors.

The credibility of the direct trust value that public neighbor node k evaluates node j is expressed as follows:

$$\partial_k = \frac{\sum_{u=1}^{q} \left| DT_{uj}^{t} - DT_{kj} \right|}{q}$$

If the value of ∂_k is larger, the direct trust value provided by node k is more likely to be malicious from malicious nodes. Therefore, the credibility threshold is set to filter out the direct trust value of $\partial_k > \partial_0$.

The credibility threshold ∂_0 is the predetermined value associated with the particular network environment and information.

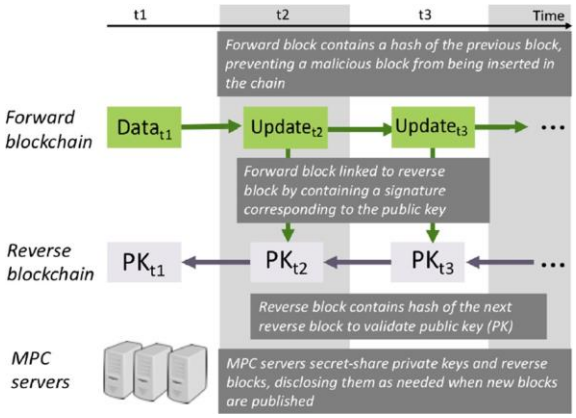
4. THE PROPOSED FRAMEWORK

To address these issues, a new trust-based, energy-efficient routing protocol was developed. Existing protocol calculates comprehensive trust value using adaptive direct, indirect, and energy trust values, making it immune to black hole, selective forwarding, sinkhole, and hello flood assaults. In addition, the adaptive penalty

mechanism and the volatilization factor are employed to quickly identify malicious nodes. Furthermore, the nodes only need to calculate the direct trust value, while the Sink obtains the indirect trust value, reducing the amount of energy consumed by repetitive calculations. Finally, the cluster heads use the complete trust value to find the safest multi-hop paths, preventing wormhole attacks. This minimizes network energy usage, speeds up the detection of rogue nodes, and resists all typical attacks, according to simulation results.

Proposed BidirecĒional Blocfi-Chain Based RouĒing in WSN

Bidirectional Blockchain combines the advantages of multiparty computing (MPC) with blockchain to protect the anonymity of keys used to sign distributed data, as well as the correctness and integrity of the data received. The forward blockchain contains the data signed by its owner, as well as the hash digest of the previous block (as in a normal blockchain), and the reverse blockchain contains the public keys used to verify the authenticity of the data stored in the forward blockchain, as well as the hash digest of the subsequent block (as in a normal blockchain). Clients can verify the validity of a public key without consulting a trusted third party by employing hash sequences that link blocks that run forward and backward in time. MPC servers keep the private keys (along with the reverse blocks) on behalf of the data owners and reveal them when necessary. The main entities and their primary goals are summarized in the below figure.



Worfiing

The proposed protocols address the security issues in WSN by utilizing a bidirectional blockchain, which consists of two separate blockchains.

1: Each block after the first contains the hash digest of the previous block (as in a normal blockchain).

2: Each block before the last contains the hash digest of the next block (as in a reverse blockchain).

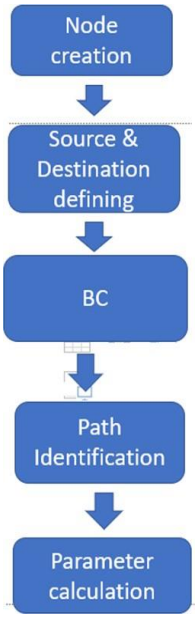
When the current reverse blockchain reaches its end and requires an extension, an additional key is secretly shared across the MPC servers - the corresponding public key is contained in the last reverse block - to sign the first block of the new reverse blockchain.While this signature is done in a distributed fashion, as with the third method above, it only needs to be done once in a while because each reverse blockchain can be any length.The forward and reverse blockchains contain the same public keys (PK), linking them to form a bidirectional blockchain. The forward blockchains store the actual data to be received or consumed by the clients. Since the longest blockchain in existence is

taken to be the correct chain, the forward blockchain ensures that data recipients have the correct, current data. The reverse blockchain stores the public keys used by the clients to verify the authenticity of the data stored in the forward blockchain. Consequently, clients can ensure that a public key is valid without referring to a TTP. Furthermore, the public keys are kept hidden until needed, securing against a scenario in which a vulnerability is found in the public-key cryptography scheme that makes it faster for an adversary to compute a private key from a public key; the window of time in which a public key is both valid and visible to nodes is narrow, making the private-key computation time narrow as well. Thus, the combination of forward and reverse blockchains into a bidirectional blockchain provides a higher degree of security compared with a traditional blockchain or PKI scheme.

To construct and maintain them, our protocol involves five distinct groups of entities: Key generators, MPC servers, signing servers, blockchain nodes, and clients. The key generator creates public-private key pairs and the reverse blockchain, then distributes private key and reverse block shares to MPC servers. MPC servers keep track of secret keys and reverse blocks and, when necessary, reveal them to a signature server.

It is the responsibility of signing servers to create new blocks and distribute them to blockchain nodes. Blockchain nodes hold the whole reverse and forward blockchains and deliver newly created blocks to clients in real time. Clients are the final consumers of the data stored in the bidirectional blockchain, and they might be any form of distributed device that receives software updates from the blockchain nodes, such as cell phones, vehicles, or aircraft.

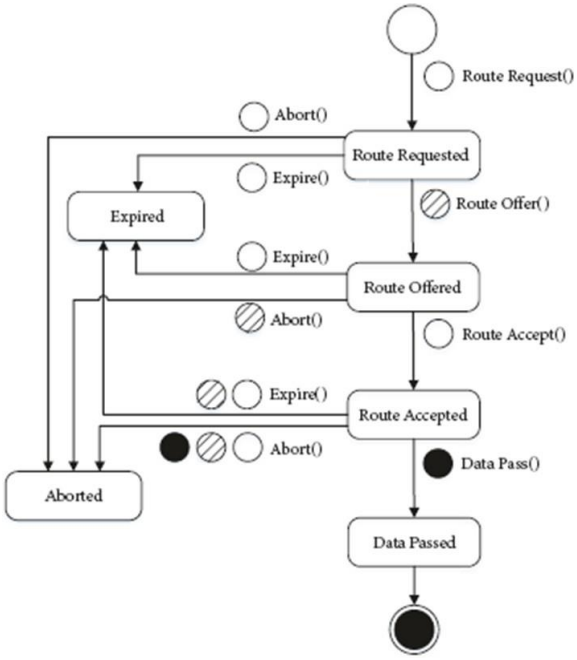
When either a client or blockchain node notices a fork (i.e., two conflicting updates of the same length), they report to the fork server the two conflicting blockchains, and the fork server may contact the MPC servers and a signing server so that a new block can be issued to resolve the fork. The newly generated block will create a blockchain that is longer than either of the current blockchains, resolving the fork. The overall Proposed block chain based new routing algorithm, is shown in the below figure.



Unlike traditional secure routing protocols, which require a central authority (CA) to assist device identification and authentication, the BCR protocol is distributed and does not require a CA. Within heterogeneous IoT networks, the BCR protocol uses smart contracts to find a path to a destination or data gateway. A path from a source device to a destination device

or gateway can be guaranteed by any intermediary device.

In a network of 14 devices, we compare the performance of BCR and the Ad-hoc On-Demand Distance Vector (AODV) routing protocol. The results reveal that the BCR protocol has a 5 times reduced routing overhead than AODV, but at the cost of a slightly worse packet delivery ratio. Both Blackhole and Greyhole attacks are fairly resistant against BCR.



RouĒe RequesĒed

When a source Device needs to reach a gateway, it creates a smart contract within the blockchain and sends the smart contract address to its neighbors. It also sets the state field within the smart contract to Route Requested. The source Device transfers some of its own blockchain tokens as a bond to a smart contract address to create a smart contract.

The possibility of earning tokens encourages intermediary Devices to respond to the route request (□□□□□ □□□□□□□ □□□□). The source Device also specifies the period for which the state of the route request within a smart contract is valid (□□□□ □□□□□□□ □□□□□□). This smart contract is termed the original contract. The below figure represents the **Route Requested Algorithm**.

```
1: function ROUTE REQUEST(DESTINATION, RRB, RRE, BLACKLIST, PARENTADDRESS(OPTIONAL),
HOP(OPTIONAL) )
2:    Transfer Gas tokens from the function caller to the block producer.
3:    Transfer RRB tokens from the function caller to the current contract address
4:    Set RRE to Route_Request_Expiry
5:    Set Blacklist to Blacklisted_Addresses
6:    if this is an original smart contract then
7:       Set Hop to 0
8:    end if
9:    if this is an intermediary smart contract then
10:      Set Hop to Hop
11:      Set Parent_Contract to ParentAddress
12:   end if
13:   Set Timestamp to Now
14: end function
```

RouĒe Offered

Each neighboring Device, which has a valid route entry to a gateway and would like to participate in relaying data packets (□□□□ Offer □□□□□□□), can respond to an original smart contract. The intermediary Device offers its services to the source device by calling on a function within the original contract and transferring some of its own tokens to the smart contract address (□□□□□ Offer □□□□).

A maximum of 3 route offers from different intermediary Devices can be stored in each contract. If the neighboring intermediary device is unaware of a route to the data gateway or destination, it can still participate in relaying data packets by creating a new smart contract, namely, the intermediary contract. The intermediary contract stores the address of the originally issued smart contract or another intermediary contract in the □□□□□□ □□□□□□□□ parameter. The figure shows the **Route offered Algorithm**.

```
1: function ROUTE OFFER(ROB, ROV)
2:    Transfer Gas tokens from the function caller to the block producer.
3:    if the function caller address is not in Blacklisted_Addresses and the number of offers is less
than three then
4:       Transfer ROB tokens from the function caller to the current contract address
5:       Set ROV to Rout_Offer_Validity
6:    end if
7: end function
```

RouĒe AccepĒed

The source Device determines whether to accept an offered route to send its data packets. It selects the next neighbor to reach a gateway based on its own internal policies. It can choose a low cost route offered by one of its neighbors or multiple neighbors to act as a relay(s) in order to increase the security and throughput of data packets.

```
1: function ROUTE ACCEPT(INTERMEDIARY)
2:    Transfer Gas tokens from the function caller to the block producer.
3:    if the function caller is Source then
4:       Move the intermediary to Selected_Route
5:       Transfer the ROB tokens of the other intermediary devices back
6:    end if
7: end function
```

AborĒed

Each device in the network can invoke the Abort function inside the smart contract at any time to stop the routing operation. The smart contract Abort function, on the other hand, functions with the caller Device type and the smart contract's present state.

```
1: function ABORT()
2:    Transfer Gas tokens from the function caller to the block producer.
3:    if state is Route Requested and the function caller is Source then
4:       Transfer Route_Request_Bond tokens back to the function caller
5:    end if
6:    if state is Route Offered then
7:       if the function caller is Source then
8:          Transfer Route_Request_Bond tokens back to the function caller
9:          Transfer Route_Offer_Bond tokens of all intermediary devices back to them
10:      end if
11:      if the function caller is Intermediary then
12:         Transfer Route_Offer_Bond tokens back to the function caller
13:      end if
14:   end if
15:   if state is Route Accepted then
16:      if the function caller is Intermediary or Destination then
17:         Transfer Route_Request_Bond of the Selected_Route and Route_Request_Bond tokens
back to Source
18:      end if
19:      if the function caller is Source then
20:         Transfer Route_Request_Bond of the Selected_Route and Route_Request_Bond tokens
back to Intermediary
21:      end if
22:   end if
23: end function
```

Expired

As the BCR protocol has various timers, a device can request that the Expire function inside a smart contract to review the timers and take action accordingly. The figure shows the **Expire** algorithm.

```
1: function EXPIRE()
2:    Transfer Gas tokens from the function caller to the block producer.
3:    if state is Route Requested then
4:       if current time is more than Route_Request_Expiry then
5:          Transfer Route_Request_Bond tokens back to Source
6:          Transfer Route_Offer_Bond tokens back to Intermediary
7:       end if
8:    end if
9:    if state is Route Offered then
10:      if current time is more than Route_Offer_Validity then
11:         Transfer Route_Request_Bond tokens back to Source
12:         Transfer Route_Offer_Bond tokens back to Intermediary
13:      end if
14:   end if
15:   if state is Route Accepted then
16:      if the function caller is Intermediary or Destination then
17:         Transfer Route_Request_Bond and Route_Offer_Bond tokens to Source
18:      end if
19:      if the function caller is Source then
20:         Transfer Route_Request_Bond and Route_Offer_Bond tokens to Selected_Route
21:      end if
22:   end if
23: end function
```

The BCR protocol does not require a central authority to authorize, add, or remove devices, or a secret key sharing mechanism as required by traditional centralized routing protocols. We evaluated the performance of our proposed protocol compared to the AODV using extensive experiments. Our results show that the BCR reduces the routing overhead by a factor of 5 compared to the AODV. It is also resistant to Greyhole and Blackhole attacks. The proposed routing protocol can also be applied to ad-hoc networks.

5. EXPERIMENTAL RESULTS

In this subsection, the suggested technique's evaluation was done using MATLAB and compared to earlier techniques.

Energy ConsumpĒion

It is a metric for the entire amount of energy usage across all nodes in a network.

Every node's energy consumption is determined based on sending, receiving, and idle energy. The total energy consumed is proportional to the number of packets sent.

$$T = \Sigma C_e \quad e$$

where T_e stands for total energy utilized and C_e stands for energy of all nodes.

NeĒworfi Delay

It is a measure for time taken from initially sent by the sender node, and to reach the destination node successfully. The packets for processing will be queuing based on this delay.

$$End\ to\ end\ delay = \left(\frac{\sum received\_packet - sent\_packet}{\sum received\_packet}\right) * 100$$

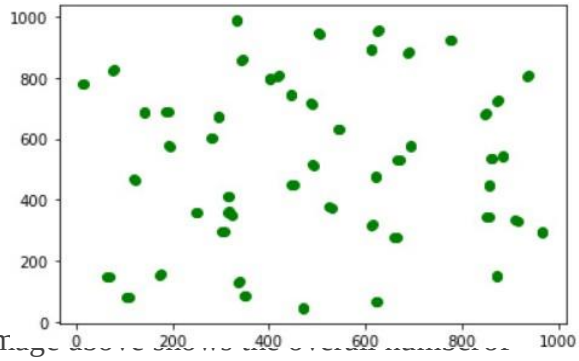
where $\Sigma_{received\_packet}$ denotes the no. of received packets and Σ_{sent_packet} denotes no. of sent packets.

NeĒworfi ThroughpuĒ

It is the number of packets collected at the receiving node per second. For better productivity and effectiveness, the network'sthroughput should be increased.

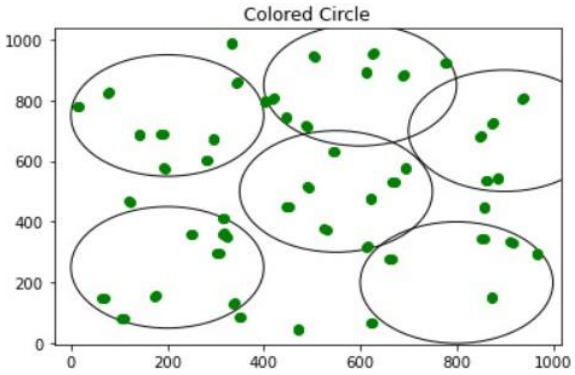
SimulaĒed ResulĒs

The following figures show the simulated result of both the proposed andthe existing methodologies.



The image above shows the overall number of living nodes obtained for various rounds.

The proposed approach outperforms other current algorithms in terms of the count of alive nodes available in the overall area as the number of rounds increases.

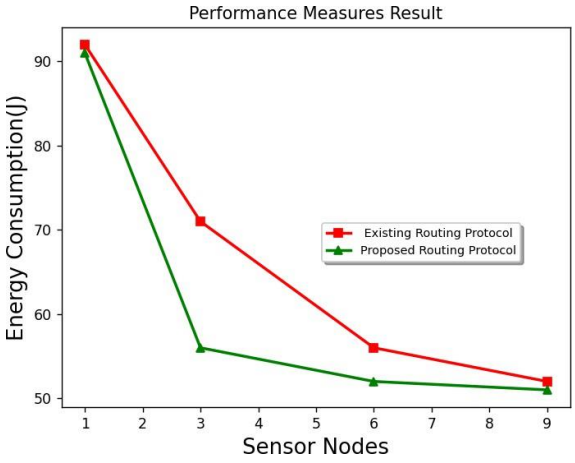


The nodes are clustered by designating a round, which may increase the network lifetime. The nodes are arranged into clusters, each with a cluster leader; the remaining nodes become cluster members. To create clusters, all nodes send a packet to the sink node and keep sending till it is successfully received.

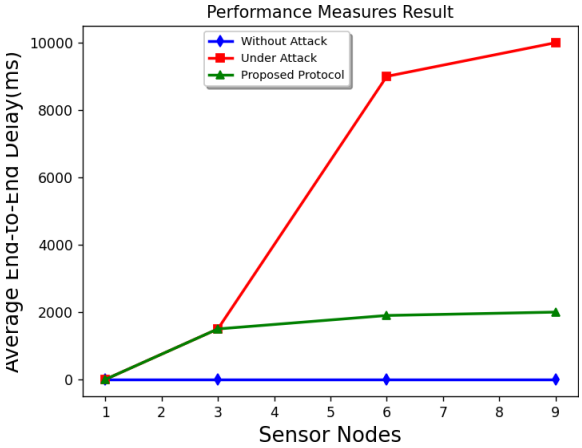
```
+--Found a route to 41 for RREP from 1 to 0
SensorNode 7 received a RREQ from 36 which is 1 -> 0
SensorNode 18 received a RREQ from 36 which is 1 -> 0
SensorNode 18 received a RREQ from 38 which is 1 -> 0
SensorNode 22 received a RREQ from 38 which is 1 -> 0
SensorNode 41 received a RREQ from 38 which is 1 -> 0
SensorNode 13 received a RREQ from 41 which is 1 -> 0
SensorNode 18 received a RREQ from 41 which is 1 -> 0
SensorNode 22 received a RREQ from 41 which is 1 -> 0
SensorNode 5 received a RREQ from 42 which is 1 -> 0
SensorNode 25 received a RREQ from 42 which is 1 -> 0
SensorNode 47 received a RREQ from 42 which is 1 -> 0
SensorNode 48 received a RREQ from 42 which is 1 -> 0
SensorNode 44 received a RREQ from 46 which is 1 -> 0
SensorNode 5 received a RREQ from 47 which is 1 -> 0
SensorNode 25 received a RREQ from 47 which is 1 -> 0
SensorNode 5 received a RREQ from 48 which is 1 -> 0
SensorNode 25 received a RREQ from 48 which is 1 -> 0
Iteration: 1
12 DATA 1 -> 0 num: 2
SensorNode 21 received a RREQ from 2 which is 0 -> 1
SensorNode 30 received a RREQ from 2 which is 0 -> 1
SensorNode 40 received a RREQ from 2 which is 0 -> 1
SensorNode 46 received a RREQ from 2 which is 0 -> 1
SensorNode 29 received a RREP from 5 which is 1 -> 0
```

The di
which
towards another. The optimal channel for data transmission has been discovered using this proposed strategy, which does not result in a significant drop in

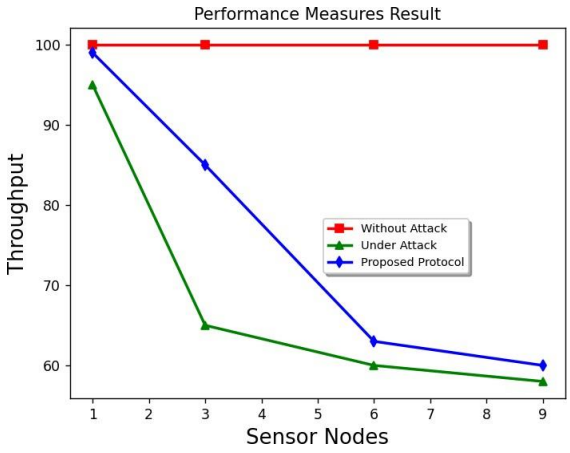
energy efficiency. When a participant desires to add a new piece of information to the blockchain, they must symmetrically encrypt it with the secret key. The transaction would then be published to the blockchain with the encrypted data.



The communication energy of a node is used to determine the indirect trust value in various contexts, as shown in the diagram above. Because the suggested technique updates and calculates the indirect trust value using the Sink (of limitless energy and powerful features), which shares the node's responsibility and saves energy. Also, as the node count of the network grows, so does their energy usage. The performance of the proposed approach, however, is still superior to that of earlier methods.



As the number of rogue nodes grows, so does the average end-to-end delay. Due to significant packet loss, routing stability of the proposed technique reduces sharply as the number of malicious nodes grows, increasing the packet delay to destination. Despite the fact that earlier techniques use a trust evaluation model, the volatilization factor evaporates past trust value and the penalty coefficient punishes harmful behaviour. As a result, the proposed approach has a shorter latency than the previous method.



The throughput is found higher for the proposed protocol than the other existing algorithms. However, existing methods fail to attain an effective result on energy consumption. Due to this the network lifetime is also reduced. To avoid such defects, the proposed protocol is maximizing the effectiveness of the entire network.

6. CONCLUSION & FUTURE WORK

We provide a secure authenticating and routing mechanism for WSNs in this project. The goal of our proposed mechanism is to perform sensor node authentication and provide secure communication between the nodes and the Base Station. The suggested

routing system chooses the nodes based on their proximity to the Base Station. The blockchain, on the other hand, is used to provide a secure authentication mechanism for nodes. Our proposed model improves the packet delivery ratio and network longevity, according to the results. The proposed approach will be tested in the future on larger networks and in a realistic routing environment.

REFERENCES

1: Y. Hu, M. Dong, K. Ota, et al. "Mobile Target Detection in Wireless Sensor Networks with Adjustable Sensing Frequency, "IEEE System Journal, Doi: 10.1109/JSYST.2014.2308391, 2014

2: M. Dong, K. Ota, A. Liu, et al. "Joint Optimization of Lifetime and Transport Delay under Reliability Constraint Wireless Sensor Networks," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 1, pp. 225-236, 2016

3: S. He, J. Chen, F. Jiang, et l. "Energy provisioning in wireless rechargeable sensor networks," IEEE transactions on mobile computing vol. 12, no. 10, pp. 1931-1942, 2013

4: C. Zhu, H. Nicanfar, V. C. M. Leung, et al. "An Authenticated Trust and Reputation Calculation and Management System for Cloud and Sensor Networks Integration," IEEE Transactions on Information Forensics and Security, vol. 10, no. 1, pp. 118-131, 2015

5: A.Liu, M.Dong, K.Ota, et al."PHACK : An Efficient Scheme for Selective Forwarding Attack Detecting in WSNs," Sensors, vol. 15, no. 12, pp. 30942-30963, 2015

6: A. Liu, X. Jin, G.Cui, Z. Chen, "Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network,"Information Sciences,vol. 230, pp.197-226, 2013

7: Z. Zheng, A. Liu, L. Cai, et al."Energy and Memory Efficient Clone Detection in Wireless Sensor Networks, "IEEE Transactions on Mobile Computing.vol. 15, no. 5, pp.1130-1143, 2016

8: P. Zhou, S. Jiang, A. Irissappane, et al. "Toward Energy-Efficient Trust System Through Watchdog Optimization for WSNs," IEEE Transactions on Information Forensics and Security, vol. 10, no. 3, pp. 613-625, 2015

9: S. Shen, H. Li, R. Han, et al. "Differential game-based strategies for preventing malware propagation in wireless sensor networks," IEEE Transactions on Information Forensics and Security, vol.9, no. 11, pp. 1962-1973, 2014

10: J. Long, A. Liu, M. Dong, et al. "An energy-efficient and sink-location privacy enhanced scheme for WSNs through ring based routing," Journal of Parallel and Distributed Computing, vol. 81, pp. 47-65, 2015

11: Yin, H., Yin, Z., Yang, Y., & Sun, J. “Research on the Node Information Security of WSN Based on Multi-Party Data Fusion Algorithm. IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)”, 2018