International Journal of Business Data Communications and Networking
Volume 17 • Issue 1 • January-June 2021

Decision Tree Trust (DTTrust)-Based Authentication Mechanism to Secure RPL Routing Protocol on Internet of Battlefield Thing (IoBT)

Prathapchandran Kannimuthu, Karpagam Academy of Higher Education, India
https://orcid.org/0000-0002-5125-1972

Janani Thangamuthu, Karpagam Academy of Higher Education, India

ABSTRACT

Providing security on the internet of battlefield things (IoBT) is a crucial task because of various factors such as heterogeneous, dynamic, and resource-constrained devices. Besides, authentication is essential, and it ensures the initial level of security in the network; therefore, ensuring authentication of various interconnected battlefield sensors/devices is the primary attention for the military applications. With this idea in mind, in this paper, a trust model that uses a decision tree to identify and isolate the misbehaving battlefield thing in the IoBT environment is proposed. The decision tree is the predictive modeling and machine learning technique that provides an accurate estimation for selecting authenticated nodes in IoBT by addressing the rank attack by the way security of IoBT environment can be ensured. The mathematical model shows the applicability of the proposed work. The simulation results show the proposed model is better than the existing routing protocol for low power lossy network (RPL) and the protocol which is similar to the proposed one.

KEYWORDS

Authentication, Decision Tree, Internet of Things, IoBT, Rank Attack, RPL, Security, Trust

1. INTRODUCTION

Internet of Things (IoT) makes considerable attention in both application domains and academic research because of its unique characteristics. It is an interdisciplinary framework in which things surrounding us are associated with the internet to provide smart and efficient services. Many of the application domains use IoT to offer new services or enhance the efficiency of the existing services (Samie et al., 2016). Such applications are transporting, environmental monitoring, e-health, industrial monitoring, smart agriculture, public safety, military application, etc (Sung et al., 2012). When designing IoT applications, some of the key characteristics should consider managing with additional

DOI: 10.4018/IJBDCN.2021010101

challenges that are constrained resources in IoT devices, an extremely large volume of data collected from various applications, and distributed network environments (Sheng et al., 2015).

Implementing the IoT concept in a battlefield environment may import numerous advantages, and can perform a higher level of operational efficiency. Modern military uniforms, battlefield objects, and weapon systems are highly equipped with sensor nodes that can aggregate and deal with the data on the place of the authorized objects and their surroundings. Incorporation of all the things within the IoT infrastructure gives an enormous amount of information for context-awareness applications (Glowacka et al., 2015). On the Internet of Battlefield Things (IoBT), the intelligence devices (Things) occupy the world of military battles where devices can interact with each other that helps armed forces on the battlefield. In the next few decades, IoBT becomes a primary existence that is widely occupied by the different categories of objects on the battlefield, many of these objects are too smart and few are only normal. Battlefield things perform a wide range of tasks including communicating, sensing, and collaborating. These things include weapons, sensors, robots, vehicles, and human-wearable devices. The task of these devices involves collecting and processing specific information, acting as agents to assist sense-making, attempt to coordinate defensive actions, and discharge different types of action on the adversary. These can be achieved collaboratively, all the things on the battlefield regularly interact, coordinate, consult, plan, and execute their actions.

In the real-time implementation, it faces a unique set of challenges including heterogeneous, highly dynamic, and largely unpredictable environment, collecting and processing data, restricted resources, collaboration, security threat from an adversary, the trustworthiness of the nodes, etc (Kott et al., 2016). Wireless sensor nodes used in a restricted environment like battlefield networks are highly vulnerable to various attacks (Jaitly et al., 2017). The adversary may access the communication channel between the sender and receiver, then modify or drop the data packets that are transferred to this communication channel (Bhushan et al., 2017). Therefore, providing security in such a network is a primary concern (Jaitly et al., 2017).

Providing standard security services such as confidentiality, authentication, integrity, authorization, non- reputation and availability are the considerable obstacles for the node's deployment in the Battlefield Environment. Among these services, the authentication of different heterogeneous entities is the main concern for a military-based application that needs to be addressed.

Every single object in the IoT network can authenticate and validate each other in the network. In IoT, authentication is defined as the ability to protect data and restrict it only to the appropriate permissions (Varshney et al., 2019). Group communication systems in the IoBT require teamwork and cooperation to accomplish the mission. It mainly depends on the trust communication among the team members (Ing-Ray Chen, 2010). Traditional cryptography-based authentication mechanisms cannot be adopted in IoT devices because of its restricted resources including memory, processing, battery power, etc. Thus, a trust-based solution plays an important role to identify the malicious nodes/devices in the battlefield environment. Therefore, the proposed model provides a trust-based solution in the IoBT environment which ensures the secure Routing Protocol for Low-Power and Lossy Networks (RPL) and isolates the malicious nodes (Rank attacker), by isolating rank attacker authentication can be achieved in the battlefield environment.

1.1 Contribution

The primary contributions of the DTTrust model are listed as follows:

- Presented the fundamental introduction for Decision tree and C4.5 algorithm, and also explains how this algorithm classifies the Things behavior based on the experience and how they determine the future behavior of the Things in the network.
- Discuss the overview of the RPL and the impact of the Rank attack on the RPL.
- Decision Tree Trust (DTTrust)-Based Authentication Mechanism is proposed.
- The mathematical model has proven the proposed model.

- The performance evaluation of the DTTrust model is compared with the existing similar work under the Rank attack to show the merits of the proposed model.

1.2 Organization

The rest of the paper has organized as follows. Section 2 presents a review of the literature. The background on the RPL protocol and Decision Tree algorithm discuss in section 3. Section 4 describes the proposed trust model, section 5 presents the mathematical analysis of the proposed model, section 6 presents the performance evaluation of the proposed trust model with various performance metrics. The conclusion of the paper present in section 7.

2. RELATED WORK

With the growing trend in the field of IoT, there has been a lot of research work that provides the solution for both security and trust in general IoT and also in IoBT. This section presents a summary of an existing research work on trust evaluation schemes on IoT that includes cryptography-based trust models and reputation-based trust models.

The authors (Glowacka et al., 2015) proposed a trust-based cognitive mechanism for military applications in the IoT network. The entities of this model should know the environmental situation, with this awareness it takes the necessary actions for identified threats. Trust is characterized as a stage of faith, depends on the direct observation and received recommendations; trust is assigned to each object. Each object monitors the communication processes within their surroundings and collects the recommendations from other trusted nodes. This model makes use of the inference method to categorize and detect the threat and also take action for a detected threat. The authors (Lahbib et al., 2017) present a Link reliable and Trust aware RPL routing protocol((LT-RPL). This system ensures trust among objects and also guarantees Quality of Service (QoS) during the RPL construction and maintenance. This model uses a multidimensional approach for an exact trust computation that considers both node trust and link trust. Trust related information is collected based on entity behavior and link quality. The trust manager computes node and link related trust. This model considers both direct observation and recommendation from its neighbor entities.

The authors (Khan et al., 2017) proposed a new network-based model to improve trust in the IoT environment. This model evaluates the trust, based on the number of interactions among the objects in the network. When packets are exchanged between two nodes, the node may gain the experience with positive or negative interactions of the other nodes in the IoT network. In this model, every object calculates the direct trust of its neighbor node, a centralized node collects this information from all nodes and provides a rating for every node in the network. If this rating falls under the threshold level then the detected object is suspicious behavior. The authors (Bhalaji et al., 2019) proposed a trust-related model for RPL to mitigate black hole attacks. They build the trust framework for RPL against black hole attacks that are implemented in both intra-DODAG and inter-DODAG levels. The authors (Airehrour et al., 2018) proposed a SecTrust system that uses the node's trust value to select the optimal routing decisions. They isolate the malicious nodes using the trust value which is computed based on the correct data packet share between the objects. It decides the objects' reliability based on data transfers to neighbor nodes in the RPL network. This model uses malicious node identification and isolation methods that identify and isolate malicious objects and increase the throughput.

The authors (Mehta et al., 2018) proposed a novel framework for IoT routing protocol against wormhole and gray hole attacks. Each node in the network monitors their neighbor node's behavior and also checks whether the nodes follow the pattern of RPL protocol or they depart from it. This model computes total trust from the direct trust and indirect trust, then the nodes are placed in the descending order and that is entrenched in RPL objective function with the ETX and Rank to transfer their data packets through the trusted nodes only and misbehaving nodes are isolated from the IoT

network. The authors (Verma et al., 2019) used ensemble learning methods for Intrusion Detection System modules which provide benefits in the kind of classification problems. In this model, the authors used the RPL-NIDDS17 training dataset to train and predict classifiers which hold the traces of routing attack on RPL protocol. The training phase has two-step, in the first step, the dataset is pre-processed, then trained using ensemble classifiers in the second step. The classifiers used in this model are Bagged, Boosted, RUSBoosted, and subspace Discriminant trees. The main reason for selecting this classifier is it can handle all types of data set (balanced, imbalanced). In the testing phase, the model returns the output of the test instance as an attack or normal class. The author concluded that with the use of ensemble learning, the performance of the model improved and assist to protect the RPL network from several routing attacks like a sinkhole, black hole, Sybil attack, clone ID attack, Selective Forwarding attack, Hello Flooding attack, and Local Repair attack. The results are compared in terms of accuracy.

The authors (Mathur et al., 2016) proposed the solution to mitigate black hole attack and selective forwarding attack in Medical Wireless Sensor Networks in the IoT. They provide the solution with the cryptography hashes and also use threshold-based analysis and neighborhood watch to identify and rectify the selective forwarding and black hole attacks. The authors (Dvir et al., 2011) developed a cryptography solution for version number and rank attack. This system avoids Version Number attack and falsifies the Rank by the malicious nodes. Version number attack makes a load on energy and it consumes more energy, to provide a solution for this attack they created hash, and also the member of this chain also creates the rank chain. The authors (Perrey et al., 2016) proposed cryptography methods to protect against internal attacks like rank Spoofing and rank replay. It is based on topological authentication. They use round trip messages to validate the upward path. The child node sends an authentication message after it receives a message from its parent node. Each parent node checks the child node rank from the testing message. The upward node checks whether the child rank is higher than its rank and also checks the rank difference.

The authors (Mabodi et al., 2020) proposed a cryptography-based authentication to address gray hole attacks in the IoT network. This model first checks the trust value for the IoT nodes then finds and removes the misbehaving nodes with the help of control packets. This model has four phases, that are checking node trust in the IoT, route testing, identifying the gray hole attack, and eliminating the gray hole attack from the system. It is a hybrid technique implemented on the AODV protocol. The authors (Zhang et al., 2019) proposed a Cuckoo-RPL to counter black hole attack in smart metering. They used the cuckoo filter to form a hash table that contains all the authorized members of the AMI network. In this system, malicious nodes are avoided from the network by blocking the DIO control messages from the malicious nodes.

The proposed work differs from the existing research work mentioned above. This model used a decision tree classifier to identify the trusted(authenticated) Battlefield Thing for communication. The rank attack is identity-related, this system detects and discards the Battlefield Thing that performs a rank attack. By providing a solution for identity-related attacks, authentication can be achieved in the IoBT environment.

3. BACKGROUND

This section presents an overview of the RPL protocol and a brief description of the decision tree and C4.5 algorithm.

3.1 RPL Protocol Description

RPL is a distance-vector routing protocol and also a source routing protocol, it functions on top of the standard IEEE 802.15.4. It supports multipoint-to-point, point-to-point, and point-to-multipoint topology. RPL builds a Destination Oriented Directed Acyclic Graph (DODAG) from IoT nodes. A DODAG contains the nodes including router, host, gateway, etc. These are all arranged themselves

into a specific form of topological structure to perform routing in Low Power and Lossy Networks (LLNs). An individual IoT network consists of several parallel RPL Instances that are all running at the same time, it can be identified by the RPL Instance ID. A single IoT network also contains several DODAGs and it is identified by the DODAG ID (unique IPv6 address). The fundamental aspect of the RPL routing protocol is self-organization, auto-repairing, loop prevention and identification, transparency, and support for multiple edge routers or sinks. To construct and manage DODAG, RPL uses various types of messages including DODAG Information Solicitation (DIS), DODAG Information Object (DIO), DODAG Advertisement Object (DAO), DODAG Advertisement Object Acknowledgement (DAO-ACK). First, the DODAG construction process is accomplished in two different ways. 1) Nodes joined in the DODAG network broadcast the DIO messages to its nearby nodes. 2)Node does not receive any DIO message then it may request DIS messages to DODAG. The DODAG allows the trickle timer, the member node of the DODAG has to transfer the DAO messages to DODAG at a particular time interval. Then, the DODAG transfers the DAO-ACK messages to all other nodes in the network.

Objective Function (OF) is used to select and optimize the route between DODAG nodes. It adopts various advantages and restrictions to choose the best path, and choose the preferred parent among the various preferred choices. Each node in the network has a unique rank value with the 16 bit which shows the present place of the nodes from the border root node. This rank value is used to manage the connection between parent and child nodes and prevent loops in the network (Winter T, 2019).

The security of the RPL protocol is highly based on the changes of the RPL packets (for example DIS, DIO, and DAO) that provide replay protection, maximum confidentiality, integrity, and delay protection. However, detecting routing attacks is a primary concern which is required to concentrate on bringing appropriate security mechanisms to eliminate these attacks (Goel et al., 2019).

3.2 Decision Tree

The construction process of the decision tree is based on a greedy algorithm and a divide and conquer technique. The construction starts from the root node. First, decides the properties to test the training data. Second, partition the training data into many small sample sets depends on its testing outcome. Every sample set consists of a new leaf node. The third step is to repeat the partition process mentioned above to attain certain end conditions and it has considered as a classifier that consists of leaves, decision nodes, and arcs. Leaves represent certain classes, decision nodes represent the attributes of the data is classified, and arcs represent the choices for these attributes (Wang et al., 2009). It is an effective classifier, that is used in more application domains. It has potential merits over other methods included noise-tolerant, handling missing values, models created by DT are easily understood by users, and also low computation cost (López-Chau et al., 2013).

The advantage of the Decision Tree is that the user does not need to be familiar with the background knowledge of the learning process.

3.3 C4.5 Algorithm

It is an advanced version of the ID3 algorithm. The main advantage over the ID3 algorithm is that it can handle continuous attributes and it uses various pruning techniques to prevent over-fitting. It uses the information gain ratio for choosing testing attributes. This algorithm first produces a decision tree from the sample set then it converts into the production rules. C4.5 algorithm used an iterative "divide and rule" strategy to produce a decision tree from the sample data set (Wang et al., 2009).

The calculation of the information gain ratio is as follows:

Assume that *'S'* be a sample data set with an *'m'* data sample. The entropy of dividing the sample set *'S'* into *'m'* various classes S_i ($i = 1, 2, \cdots, m$), and each class in *'S_i'* contains *'m_i'* samples. Entropy defines the purity of a sample set.

The entropy of dividing sample set *'S'* into *'n'* classes is defined as follows:

$$Entropy\left(s\right) = \sum_{i=1}^{m} - p\left(i\right) * log_2 p\left(i\right) \tag{1}$$

where p(i) = m_i/m is the probability of a sample set (S) that belongs to class *'S_i'*.

Gain information is defined as follows:

$$Gain\left(S, A\right) = Entropy\left(S\right) - \sum_{i=1}^{j} - \frac{\left|S_i\right|}{\left|S\right|} * Entropy\left(Si\right) \tag{2}$$

where *'A'* is an arbitrary attribute in *'S'*, it has *'j'* distinct values. *'S'* can be separated into various subsets *'j'* with *'A'* {S1, S2,...Sj}.

Split information is calculated as follows:

$$SplitInfo\left(S, A\right) = -\sum_{i=1}^{j} \frac{|\ S_i\ |}{\left|S\right|} * log_2 \frac{\left|S_i\right|}{\left|S\right|} \tag{3}$$

This value represents the attribute A's information divide with *'S'* to *'j'* subset.

Gain_Ratio is computed as follows:

$$Gain_{-Ratio\left(S,A\right)} = \frac{Gain\left(S, A\right)}{SplitInfo\left(S, A\right)} \tag{4}$$

where, Gain (S, A) is the gain information of the attribute *'A'*, and SplitInfo (S, A) is the split information of the attribute. Selecting the highest rate of information gain attributes as property division (Yuan et al., 2016).

4. DECISION TREE TRUST MODEL (DTTRUST MODEL)

The proposed model used Decision Tree to construct the trust model for IoBT. In a highly distributed battlefield environment, the Battlefield Thing that successfully joined the network may change its behavior to perform its specific aim, because this node is seized by an attacker. Selecting the trust metric is important in a behavior-based trust model. Because it depicts the past behavior of the node's performance (including communication, routing, data processing, etc) and determines the node's future behavior. Battlefield Thing uses a trust metric to classify and isolate the bad performing parents and children over RPL routing protocol. Once a malicious node is isolated, that is not used for communication anymore. But for the selfish node, one chance will be given to correct its behavior. This model is mainly designed to mitigate the rank attack. In this paper, the term nodes and Battlefield Things are interchangeable.

4.1 Design of DTTrust

The design of DTTrust contributes to two components. One is trust estimation and another one is identifying node's behavior. To estimate the trustworthiness of the Battlefield Thing and identify the rank attacker, the single trust metric is not adequate to determine the node behavior. This model uses

metrics such as packet delivery ratio, packet correctly forwarded ratio, energy consumption, load, and rank value to detect the rank attacker.

4.2 Network Model Assumptions

DTTrust developed with the following underlying assumptions:

1. The Network model is based on the pure Internet of Battlefield Thing (IoBT) environment. These things are heterogeneous which have different capabilities in terms of energy, processing, memory, etc.;
2. Decentralized Network: There is no centralized trusted object on the battlefield, thus each Battlefield Thing has to maintain their neighbor node's trust value to transfer their data;
3. Restricted Resources: Battlefield Things are small in size and their memory capacity, energy also limited. It may get drained due to sensing, monitoring, updating, and processing capacity. These things can be compromised by an adversary;
4. Dynamic Topology: Battlefield Things may leave or join any network at any time;
5. Cooperation: Cooperation of Battlefield Things is very important to achieve a mission;
6. There are three types of nodes in the example network scenario that are trusted, selfish, malicious. The trusted node performs well in terms of forwarding data packets, cooperativeness, etc. A selfish node may exist in the network that is not a malicious node but it does not cooperate with its neighbor and drop the data packet to preserve its computing resources such as battery energy, CPU cycle, etc. A malicious node performs malicious activities to disrupt the main network topology. It may drop the data packets, increase the network traffic, and may modify the data packets.

4.3 Adversary Model

6LoWPAN does not authenticate the node before joining the network. Due to this, any malicious node can easily join to the network. And also, RPL devices do not have strong security and tamper-resistant ability, thus making the adversary able to seize the node and obtain the cryptography information to use it for functioning legitimately in the IoT network (Kamble et al., 2017). RPL is vulnerable to various kinds of internal attacks. Identifying these types of attacks is difficult because it acts as a trusted node. Nodes may perform different types of misbehavior activities; one is malicious behavior and another one is selfish behavior. The proposed model focuses on providing a solution for identifying the Rank attacker in a particular scenario.

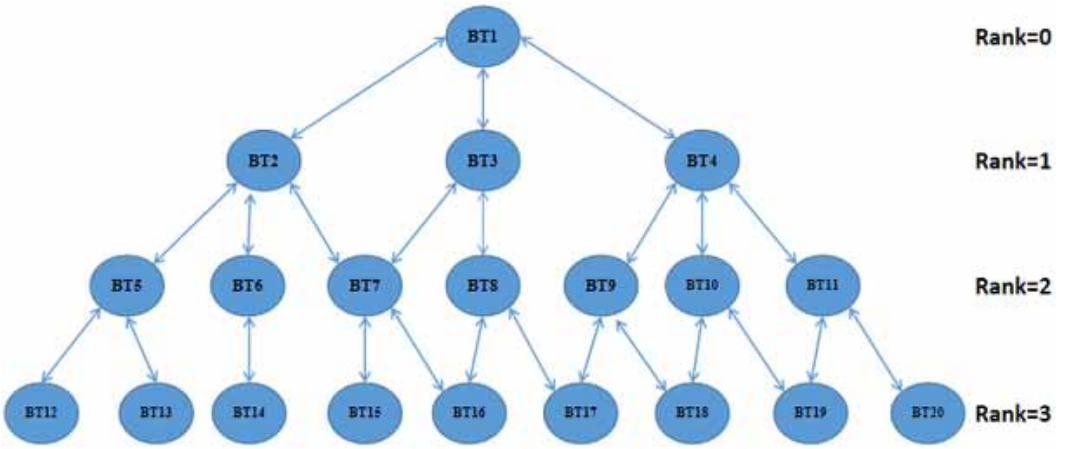
4.3.1 Rank Attack

It is the most defective one where the adversary intentionally increases or decreases the rank value. Rank is an essential parameter of RPL that allows for optimal routes and avoids loop and DODAG saturation. It may be maliciously operated by malicious nodes to make adverse effects (Glissa et al., 2016). It can be classified into two types as an Increased rank attack, Decreased rank attacks.

4.3.2 Decreased Rank Attack

The most devastating attack is a decreased rank attack where the rank attacker (misbehaving node) broadcasts a low-rank value and it leads to many parts of the DODAG that can be connected to the DODAG root through this malicious node (Glissa et al., 2016). In this model, Battlefield Things (Nodes) may be compromised by an attacker. Malicious nodes launch the decreased rank attack to attract its neighbors hence most of the neighbors selected this malicious node as its preferred parent and transfer their data through this node. Then, it combines with some other attack to drop or alter the data to interrupt the mission in the battlefield environment.

Figure 1. Example network scenario with no attacks



4.3.3 Increased Rank Attack

In an increased rank attack, the misbehaving nodes increase its rank value to show that it is far away from the root node. This node is a nonoperative node(selfish) which uses their resources and energy to transfer only its packet and does not forward others (Glissa et al., 2016). In this model, the selfish node launches this increased rank attack. This node maliciously increased its rank to pretend that it is far away from the sink node (root node). Therefore, the neighbor of this node will not select this node as the preferred parent. If any of the nodes select this malicious node as a parent node and transfer the data through this selfish node, the selfish node will not listen or transfer the data packets. The main intention of this node is to save its battery power, memory, CPU cycle, etc.

Figure 1 illustrates the simple battlefield environment with no attackers. BT represents the Battlefield Thing. All BTs are trusted and cooperatively worked to achieve the mission. Figure 2, illustrates the battlefield environment with increased and decreased rank attackers. In this example scenario, Battlefield Thing BT4 is a selfish node, it maliciously manipulates its rank to perform an increased rank attack by the way it tries to save its energy. Battlefield Thing BT6 is a malicious node,

Figure 2. Example network scenario with Rank attacks

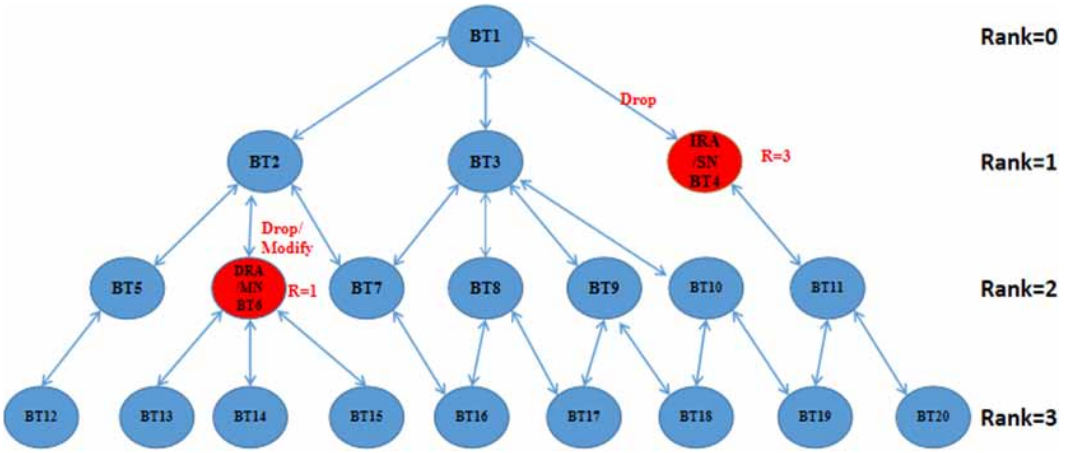


Table 1. PDR threshold table

Level	Threshold	Description
1	PDR ≤ Threshold	Bad
2	PDR > Threshold	Good

it performs a decrease rank attack to attract its neighbor and perform malicious activities like a drop or modify the data packets.

4.4 Trust Management

The main goal of this paper is to describe and analyze a complete trust model to mitigate rank attacks on the Internet of Battlefield Thing Environment. The proposed model uses either direct trust or recommendation trust. If the node has any previous interaction, it does not focus on second-hand information for trust computation because the direct experience is the most reliable source of information. If the node is new for the network, it will not have any direct experience. In such cases, it receives recommendation value from the neighbor node for its parent node. In this trust model, direct trust and recommendation trust have not aggregated because it takes extra computation. If the node does not have any previous interaction, it receives recommendation trust otherwise, it uses its own direct experience for trust computation. Trust metrics are also not aggregated, individual threshold values have been fixed for each trust metric. This model uses the decision tree to classify the node's behavior as trusted, selfish, and malicious. The decision tree can solve the classification problem. It learns the model from the trust metric set and classifies the node's behavior into one of the classes (trusted, selfish, and malicious).

4.4.1 Direct Trust

In this model, trust metrics have not aggregated to form a single trust value and each trust metric is assessed separately with its threshold value. Decision trees have been used to classify and determine the node's future behavior using these trust metrics.

4.4.1.1 Packet Delivery Ratio (PDR)

Packet delivery ratio (PDR) is measured as the ratio between the number of data packets delivered to the receiver at *'t'* time and the number of data packets sent by the source at *'t'* time (Anggoro et al., 2012).

The equation to measure the PDR as follows:

$$PDR_{i,j}\left(t\right) = \frac{TPR_{j}\left(t\right)}{TPF_{i,j}\left(t\right)} \tag{5}$$

where:

- TPR_j- the total amount of data packets received by node *j*.
- TPF_i -the total amount of the data packet forwarded by node *i*.

4.4.1.2 Packet Correctly Forwarding Ratio (PCFR)

The proportion of the correctly forwarded data packets is measured as the ratio between the total amount of correctly forwarded data packets and the total amount of data packets received. The term correctly

Table 2. PCFR threshold table

Level	Threshold	Description
1	PCFR ≤ Threshold	Bad
2	PDR > Threshold	Good

forwarded means, relay node not only forward the data packets to its neighbor but also forwards the data packets without any modification. For example, when a malicious relay node forwards the data packets after altering the data packets, then it is not recognized as correctly forwarding behavior, the forwarding ratio of the malicious node will be low (Wang et al., 2014). In the battlefield network, this metric is to ensure data reliability and it is also used to identify the decreased rank attack with data modification. The packet forwarding ratio calculated as follows:

$$PCFR_{i,j}\left(t\right) = \frac{PCF_{j}\left(t\right)}{TPR_{i,j}\left(t\right)} \tag{6}$$

- $PCF_j(t)$ denotes the total number of packets correctly forwarded by node *j* at *'t'* time;
- $TPR_{i,j}(t)$ represents a total amount of data packets received successfully from the node *i* by node *j* at *'t'* time.

4.4.1.3 Energy Consumption

Energy consumption determines the node's behavior. The node may either consume too much energy or very less energy than the actual energy. In this model, the node which performs a decreased rank attack consumes more energy due to falsely manipulates its rank to perform the decreased rank attack. It makes many of its neighbors select this node as the parent node. Due to the heavy traffic load near this malicious node, it consumes more energy than the required energy (Zhong et al., 2017). A selfish node consumes less energy than the actual energy because it drops the data packets instead of forwarding to its neighbors. The Energy consumption of node i for forwarding data packets to its neighbor as follows:

$$EC_{i}\left(t\right) = ER_{i}\left(t\right) + EF_{i}\left(t\right) + EACK_{i}\left(t\right) \tag{7}$$

where:

- ER- Energy consumption for packets received;
- EF- Energy consumption for packets forwarded;
- EACK- Energy consumption for acknowledgment.

Table 3. Energy consumption threshold table

Level	Threshold	Description
1	If TH1≤EC ≤TH2	Normal
2	If EC<TH1	Low
3	If EC>TH2	High

Table 4. Node traffic load threshold table

Level	Threshold	Description
1	NTL≤ Threshold	Light
2	PDR> Threshold	Heavy

4.4.1.4 Node Traffic Load

Data traffic is the total number of data transfers over the network at a particular time. A load balance is a technique that balances the traffic over the network. Node traffic is computed based on the total number of a child node present for every parent node in the network (Sankar et al., 2018).

The following equation measure the node traffic of the parent node based on its number of a child node:

$$NTL_i\left(t\right) = \sum_{k=1}^{n} Child\_Count\left(k\right) \tag{8}$$

where:

- NTL- Node-traffic Load
- Child_Count(k)- k number of Child for parent node i

4.4.1.5 Rank Value

In this model, every node computes its rank using the hop count objective function.

Hop Count: This routing metric is used to measure the number of hops between the source node and the destination node. In the Contiki OS, the default Objective Function is OF0 and picks the optimal path based on the lowest hop count towards the root node. Contiki OS uses 16 bit to store rank values with the units of 256 and has a maximum of 255 hops.

Each child node computes its rank using the rank of the parent node. It is calculated as the sum of parent rank value and the default_min_hop_rank_increase. The default min hop rank increase value is 256 in the RFC (6550). The rank computation based on the hop count Objective Function is computed as follows:

$$R\left(CN\right) = R\left(PN\right) + \left(default\_\min\_hop\_rank\_increase\right) \tag{9}$$

where:

- R(CN)-Child node Rank value
- R(PN)-Parent Node Rank value

default_min_hop_rank_increase=256

- Child Node (CN) select the Parent Node (PN) that reduces the rank value of the child node(R(CN)) (Abdel Hakeem et al., 2019).

4.4.2 Recommendation Trust

When a node has no experience with its parent or child node, then it receives the recommendation from its neighbor nodes. Based on the experience, neighbor nodes provide recommendation value to the requested node. Average recommendation trust is calculated from the received recommendation value. The following equation is used to compute the Recommendation Trust:

$$RT_{i,j}\left(t\right) = \frac{\sum_{k=1}^{m} RV_{k,j}\left(t\right)}{m} \tag{10}$$

where:

- m - Number of nodes provides recommendation value to node *'i'* for node *'j'*.
- $RV_{k,j}(t)$ -k^{th} node provides recommendation value for node *'j'*.

Based on the computed recommendation trust, node i decide whether to select node j or not. Once node *'i'* interact with node *'j'*, then it will use its information for further communication.

4.4.3 Detecting and Isolating Malicious Battlefield Thing in IoBT Environment

There are two types of misbehaving Battlefield Things in the battlefield scenario. One is selfish behavior, due to the constrained energy. The selfish node will not use their energy to forward the data packets of other nodes. These nodes launch an increased rank attack to pretend that it is far away from the sink node. Another one is, malicious behavior; an adversary may compromise the Battlefield Things to disrupt the network topology and the mission. To interrupt the network topology, malicious Battlefield Thing performs a decreased rank attack to attract its neighbor battlefield thing to select this node as a preferred parent. Hence, a large part of the Battlefield Thing connected to the sink node through this malicious Battlefield Thing. Later, it drops or modifies the data packets to interrupt the mission.

4.4.3.1 Parent Node Trust Calculation

Different trust metrics and their corresponding node behavior based on past observations are presented in Table 5. These metrics are used to identify the Rank attack.

In the initial stage, the decision tree is empty, then the C4.5 algorithm builds the decision tree from the root, the decision nodes and leaves nodes are added to the tree. Decision nodes are also

Table 5. Decision tree table

S.NO	PDR	PCFR	E.C	LOAD	Node Behavior	Description
1	Good	Good	Normal	Heavy	Trusted	No Attack(Nearby Root Node).
2	Good	Good	Normal	Light	Trusted	No Attack.
3	Good	Bad	High	Heavy	Malicious	Decreased Rank attack with modification.
4	Bad	Good	High	Heavy	Malicious	Decreased Rank attack with data packet drop.
5	Bad	Good	Normal	Heavy	Malicious	Decreased Rank attack with data packet drop.
6	Bad	Good	Low	Light	Malicious(S)	Increased Rank Attack.

Table 6. Gain ratio for trust metrics

Trust Metric	Gain	Gain Ratio
PDR	0.4585	0.4585
PCFR	0.1080	0.16644
LOAD	0.0445	0.084
EC	0.42875	0.3013

called internal nodes and it holds test attributes. Leaf nodes are also called terminal nodes that hold a label of the class.

Gain Ratio Calculation for Trust Metrics

Trust metric PDR Gain Ratio is computed as follows:

Gain (NB, PDR) =Entropy (NB)-Σ P(NB|PDR). Entropy (NB|PDR)

Entropy (NB) is a Global Entropy that is calculated as follows:

Entropy (NB)=-p(M)x$\log_2$p(M)-p(T)x$\log_{2\,p}$(T)=-4/6$\log_2$4/6-4/6$\log_2$4/6=0.91733
Entropy (NB|PDR=Good) = -(2/3) x$\log_2$(2/3) -(1/3) x$\log_2$(1/3) =0.91733
Entropy (NB|PDR=Bad) = -(3/3) x$\log_2$(3/3) -(0/3) x$\log_2$(0/3) =0

Information gain is calculated using equation:

Gain (NB, PDR) =0.917-(3/6x0.917) -(3/6x0) =0.4585

Split information is computed using equation:

SplitInfo(NB,PDR)= -3/6x$\log_2$3/6-3/6x$\log_2$3/6=1

Gain Ratio is computed using equation:

Gain Ratio (NB, PDR) =0.4585/1=0.4585

Gain Ratios for other trust metrics are computed using the same procedure.

Table 6 shows the information gain and gains ratio for each trust metric.

The trust metric with the greatest gain ratio is picked as the splitting attribute. Based on the Gain ratio, decision rules are constructed as follows.

Algorithm 1 (Decision Tree Rules)

```
P_Node Behavior(PDR,PCFR,EC,LOAD)
If P_Node.PDR='Good'
If  P_Node.EC='Normal'
If P_Node.PCFR='Good'
If P_Node.LOAD= 'Light'
```

```
Return "Trusted"
Else if P_Node. LOAD="Heavy'
Return "Trusted"
End if
End if
Else If P_Node.EC='High'
if P_Node.PCFR='Bad'
If P_Node.LOAD= 'Heavy'
Return "Malicious" //Decreased Rank Attack with Data modification
End if
End if
End if
Else if P_Node.PDR='Bad'
If P_Node.EC='Normal'
If P_Node.PCFR='Good'
If P_Node.LOAD='Heavy'
Return "Malicious" //Decreased Rank Attack and drop the data
packets
End if
End if
Else if P_Node.EC='Heavy'
If P_Node.PCFR='Good'
If P_Node.LOAD='Heavy'
Return "Malicious" //Decreased Rank Attack and drop the data
packets
End if
End if
Else if P_Node.EC='Low'
If P_Node.PCFR='Good'
If P_Node.LOAD='Light'
Return "Selfish" // Increased Rank Attack and drop the data
packets
End if
End if
End if
End if
```

The child node evaluates its parent node's trust using the above decision rules. If the node Behavior of the parent battlefield thing falls under the class trusted then the parent is trusted and data transferred through this node otherwise the node is malicious or selfish.

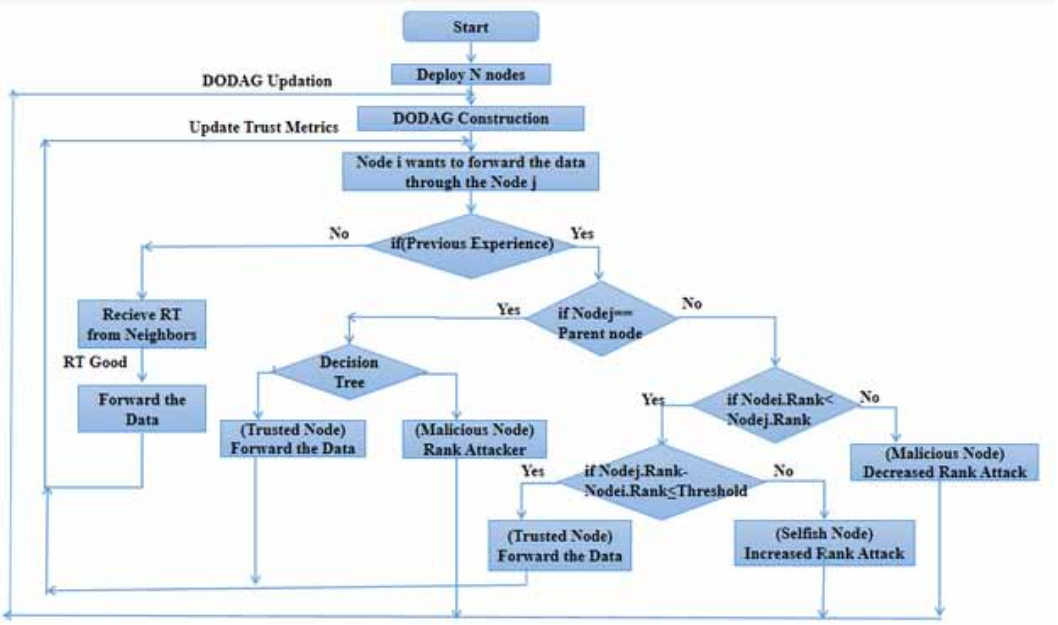
4.4.3.2 Child Node Trust Calculation

In this model, the Rank value is computed based on the hop count. The child node computes its rank using min_hop_rank_increase and parent rank value. Using the following rules, the parent node checks whether the child node is a rank attacker or not.

Algorithm 2 (Rank value checking)

```
If parent_node.Rank < child_node.Rank
If(child_node.Rank - parent_node.Rank ≤ Threshold Value)
Return "Trusted"
```

Figure 3. Overall structure of DTTrust




```
Else
Return "Selfish" //Increased Rank Attack
End if
Else
Return "Malicious" // Decreased Rank Attack
End if
Parent Battlefield Thing checks the rank value. If the child
Battlefield Thing is trusted then the data packets transfer
through its child node.
```

In both parent and child node trust calculation, when node identified the Malicious or Selfish behavior then the information about node behavior is broadcast to its neighbor Battlefield Thing, thus prevent other Battlefield Things will not transfer the data packets through this malicious or selfish Battlefield Thing and disconnect the link from that Battlefield Thing. By doing this, the rank attacker is discarded from the network and ensures authentication in the IoBT environment to effectively achieve the mission.

5. MATHEMATICAL ANALYSIS OF DTTRUST MODEL

We take the examples shown in Figures 4 and 5 for our mathematical analysis.

5.1 Parent Node Trust Calculation

The child node evaluates its parent node trust behavior based on the direct experience. To classify the node behavior, a decision tree algorithm (Algorithm 1) is used. Decision tree rules take trust metrics as parameters and return the node's behavior (Trusted, Malicious, Selfish) as output.

Figure 4. Example Network scenario with Decreased Rank Attack in IoBT Environment

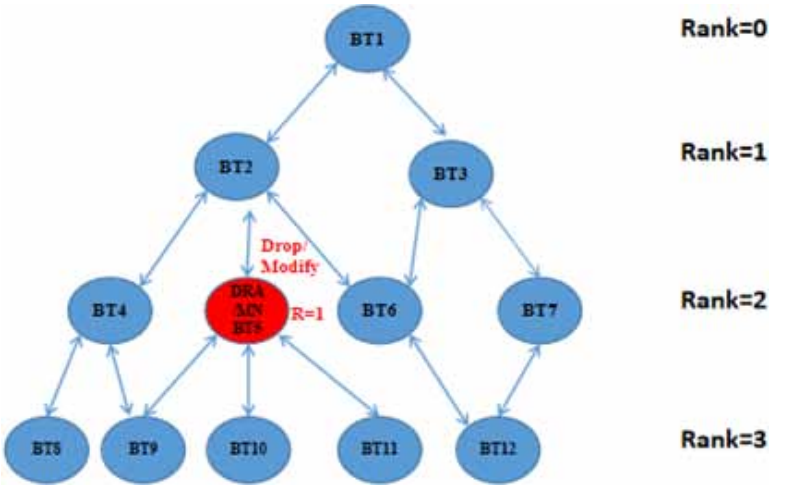
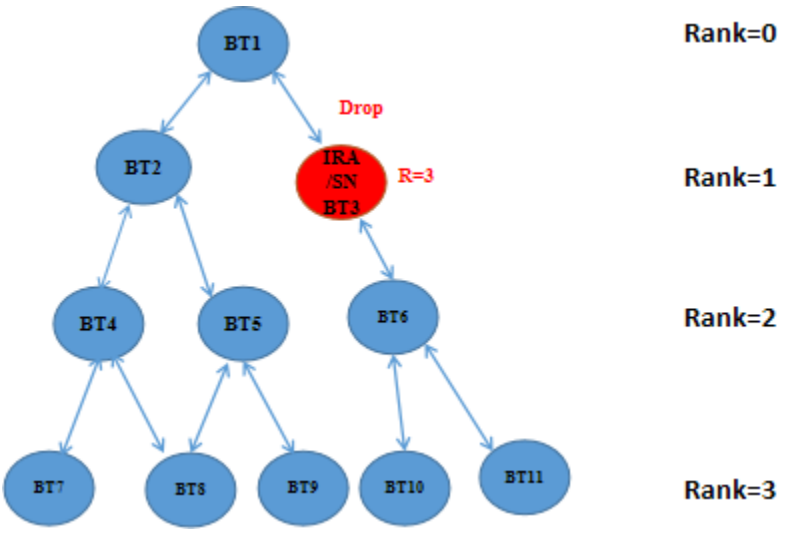


Figure 5. Example Network scenario with Increased Rank Attack in IoBT Environment



In figure 4, child node BT9 has two parent nodes BT4 and BT5. BT9 computes the trust values of BT5 and BT4 based on its experience. We assume trust metrics of BT5 as follows, PDR=Bad, PCFR=Good, E.C=High, Load= Heavy.

These trust metrics values are passed to the decision tree rules (Algorithm 1) that returned the BT5 node as malicious nodes.

We assume trust metrics of BT4 as follows, PDR=Good, PCFR=Good, E.C=Normal, Load= Light. Decision tree rules returned BT4 as trusted nodes.

Therefore, BT9 disconnects the link from the parent node BT5 and transfers its data through the parent node BT4.

In figure 5, node BT5 evaluates its parent node BT3's behavior based on its experience. We assume trust metrics of BT5 as follows, PDR=Bad, PCFR=Good, E.C=Low, Load= Light.

These trust metrics values are passed to the decision tree rules(Algorithm 1) that returned the BT6 node as a selfish node.

Therefore, BT6 disconnects the link from the parent node BT3 and selects a new trusted parent to transfer its data.

5.2 Child Node Trust Calculation

In the proposed model MinHopRankIncrease(256) was used to calculate child node rank value. For example, in figure 4, the child node(BT5) Rank should be greater than the parent node(BT2) Rank, but the Child node(BT5) maliciously manipulate its rank to launch a decreased rank attack, therefore the child node(BT5) Rank is not greater than the parent(BT2) node Rank. Parent node BT2 evaluates the behavior of its child node BT5 node using the rank value checking rules (Algorithm 2) that return the node behavior as malicious. Therefore, the parent node identifies that the node BT5 is a malicious node and disconnects the link from its child node.

In figure 5, the root node is the parent node of the node BT3. Parent node BT1 evaluates the behavior of its child node BT5 using the rank value checking rules (Algorithm 2). BT3 launched an increased rank attack and maliciously manipulated its rank. BT1 checks the BT5 rank value and identifies that node BT3 is a selfish node and it launches an increased rank attack.

Both child and parent nodes disconnect the link from the malicious nodes. Thus, malicious nodes are discarded from the battlefield environment and authentication is ensured.

6. SIMULATION RESULTS AND DISCUSSIONS

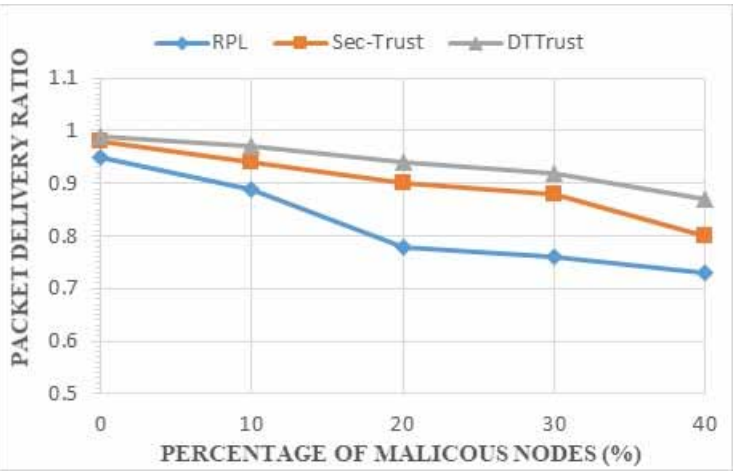
6.1. Performance Evaluation Metrics

The DTTrust model is evaluated in the Contiki 3.0 OS and the Cooja simulator. The DTTrust model uses TMote Sky (Sensor nodes) as a mote type. The following table shows the simulation parameters of the proposed trust model.

Table 7. The simulation parameters of the proposed DTTrust model

System Parameters	Reflection in Real Scenario	Values
Number of nodes	Things involved in the battlefield to achieve a mission.	50
Mote Type		TMote Sky
Simulation Time	Represents the overall time to execute the mission	3600Sec
Network Coverage Area	Represents the coverage area of the battlefield.	300mx300m
Data Rate	Represents the amount of digital data transfer from one mission point to another.	3072bps
Data Packet Size	Represents the unit of data that is originated from one mission point to another	64 byte
Traffic	Type of data that is transmitted during the mission.	UDP
Mac Layer		IEEE 802.15.4
Communication Range		50m
RPL Parameter		MinHopRankIncrease=256
Routing Protocol		DTTrust, SecTrust, RPL

Figure 6. Packet delivery ratio



6.2 Simulation Results

The performance evaluation of the DTTrust model is compared with an existing protocol SecTrust (Airehrour et al., 2018) in terms of Packet Delivery Ratio, End to End Delay, Routing overhead, and Attack detection.

6.2.1 Packet Delivery Ratio

It is a proportion of the total amount of data packets forwarded by the source node and the total amount of data packets received by the destination node. It is one of the significant metrics for evaluating the performance of the proposed model. This metric is used to analyze the delivery ratio for the individual node and also for the whole network. Protocols are evaluated by varying percentages of the malicious nodes. These malicious nodes are increased from 0 to 40%. Figure 6 depicts the packet delivery ratio of RPL, SecTrust RPL, and DTTrust. With the absence of misbehaving nodes in the IoT network, the delivery ratio of all models is relatively close to 1. The percentage of misbehaving nodes increases, the rate of packet delivery ratio of DTTrust is higher than the other schemes because the DTTrust model uses a decision tree that identifies the malicious nodes in the early stage. Therefore, nodes in the DTTrust model find alternate trusted nodes to transfer its data packets, thus increasing the delivery ratio in the proposed model.

6.2.2 End to End Delay

It is measured as an average time needed to send a packet from source to destination. It is another important metric to measure the functionality of the proposed protocols. The presence of misbehaving nodes in the IoT network increases the delay. Figure 7 depicts the impact on the delay of different protocols (RPL, SecTrust, and DTTrust) with the varying percentage of the malicious nodes. It shows that the delay of DTTrust is less than the other two protocols because DTTrust isolates the malicious nodes (Rank attacker) in the initial stage. SecTrust computes the trust values for the parent node only. The Proposed model computes trust values for both parent and child node and effectively avoids the malicious nodes from the network, therefore, the end to end delay decreased.

6.2.3. Routing Overhead

It is the proportion of the total number of route packets to the total number of data packets. Figure 8 depicts the routing overhead of the RPL, SecTrust, and DTTrust with varying percentages of malicious

Figure 7. End to End Delay

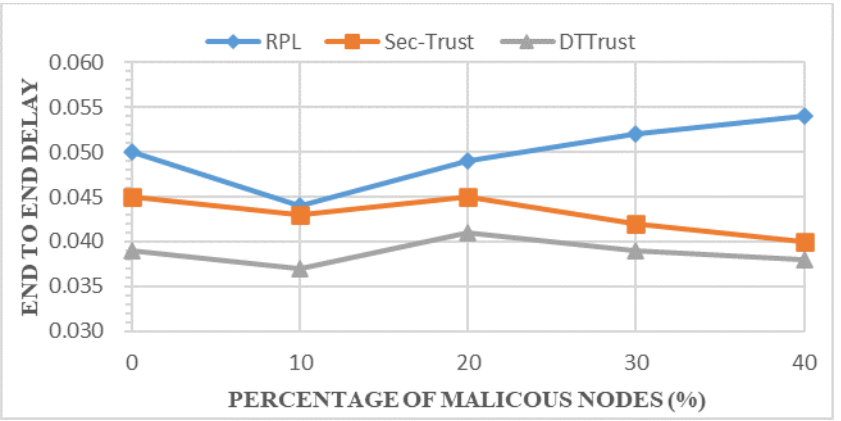
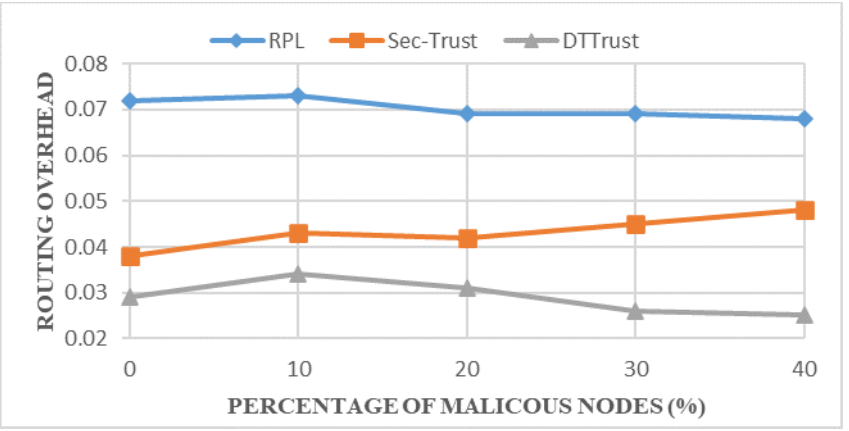


Figure 8. Routing overhead

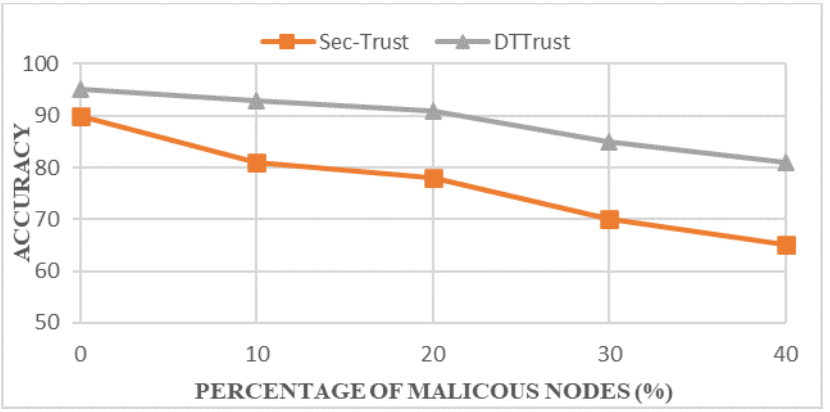


nodes. It can be noticed that DTTrust routing overhead is lower than the other two protocols. The rank attack degrades the functionality of the network. The increased rank attack creates the loop, therefore control messages increase and make an unstable network. DTTrust model identifies and avoids both increased and decreased rank attackers. Thus, the routing overhead is low in the proposed model. In SecTrust, the child node computes the parent trust value and selects the high trust parent as a preferred parent. The parent node does not evaluate the trustworthiness of the child node. Therefore, the identification of increased rank attackers is low in SecTrust, so routing overhead is high in the network.

6.2.4. Detection Accuracy

Figure 9 shows the detection accuracy of the DTTrust and SecTrust in identifying the malicious nodes (Rank attacker). The accuracy of both DTTrust and SecTrust degrades with the increasing percentage of the malicious nodes. However, the accuracy of the DTTrust model is higher than the SecTrust because the DTTrust model uses the decision tree that finds the malicious nodes with the highest accuracy. The proposed model also uses multiple trust metrics to identify the rank attacker. Whereas the SecTrust model considers only the packet forwarding ratio trust metric, therefore the detection accuracy is low.

Figure 9. Detection accuracy



7. CONCLUSION

The Internet of Battlefield Thing is one of the emerging applications to enhance mission effectiveness in the battlefield environment. However, the security in IoBT is a more challenging one, because the adversary can attack the battlefield thing and interrupt the mission. The proposed DTTrust is a behavior-based trust model to evaluate the trustworthiness among the battlefield things in the IoBT environment. The rank attack is an identity-related attack, by detecting the rank attack and discarding the malicious battlefield thing from the network can provide security and ensure authentication in the IoBT. In this model, trust is calculated for both child and parent battlefield things. In a battlefield environment, a malicious battlefield thing performs decreased rank attack to attract its neighbor, then drop or modify the data packets. Selfish battlefield thing launch increased rank attack to save its energy. The impact of these attacks is energy consumption, load, packet drop, and packet modification. Based on these impacts, the proposed model uses the trust metrics to identify the rank attacker. Trust factors used in this model are energy consumption, packet delivery ratio, packet correctly forwarded ratio, load, and rank value. The decision tree uses this trust metric to classify the battlefield things into their corresponding classes. Trusted battlefield things only involved in the mission, while malicious battlefield things are effectively prevented from the network, thus ensuring the authentication in the IoBT environment. The proposed trust model has been embedded in RPL and the performance of the DTTrust is evaluated using a cooja simulator. The performance evaluation shows the effectiveness of the DTTrust with varying percentages of malicious nodes as compared to SecTrust.

ACKNOWLEDGMENT

This research work is supported by Karpagam Academy of Higher Education (Deemed to be University), Coimbatore – 641 021, Tamilnadu, India, through a Seed Money Project.

Conflicts of Interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

Process Dates:

Received: April 7, 2020, Accepted: September 15, 2020

Corresponding Author:

Correspondence should be addressed to Prathapchandran Kannimuthu, kprathapchandran@gmail.com

REFERENCES

Abdel Hakeem, S., Hady, A., & Kim, H. (2019). RPL Routing Protocol Performance in Smart Grid Applications Based Wireless Sensors: Experimental and Simulated Analysis. *Electronics (Basel)*, *8*(2), 186. doi:10.3390/electronics8020186

Airehrour, D., Gutierrez, J. A., & Ray, S. K. (2018). SecTrust -RPL: A secure trust-aware RPL routing protocol for Internet of Things. *Future Generation Computer Systems*. Advance online publication. doi:10.1016/j.future.2018.03.021

Anggoro, R., Kitasuka, T., Nakamura, R., & Aritsugi, M. (2012). Performance Evaluation of AODV and AOMDV with Probabilistic Relay in VANET Environments. *2012 Third International Conference on Networking and Computing*. doi:10.1109/ICNC.2012.47

Bhalaji, N., Hariharasudan, K. S., & Aashika, K. (2019). A Trust Based Mechanism to Combat Blackhole Attack in RPL Protocol. *ICICCT 2019 – System Reliability, Quality Control, Safety, Maintenance and Management,* 457–464. doi:.10.1007/978-981-13-8461-5_51

Bhushan, B., Sahoo, G., & Rai, A. K. (2017). Man-in-the-middle attack in wireless and computer networking — A review. *2017, 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)*. doi:10.1109/icaccaf.2017.8344724

Chen, I.-R. Mission-Dependent Trust Management in Heterogeneous Military Mobile Ad Hoc Networks, *Proceedings of the 15th International Command and Control Research and Technology Symposium (ICCRTS '10)*.

Dvir, , Holczer, & Buttyan. (2011). Vera - version number and rank authentication in rpl. *IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*. doi:10.1109/MASS.2011.76

Glissa, G., Rachedi, A., & Meddeb, A. (2016). A Secure Routing Protocol Based on RPL for Internet of Things. *2016 IEEE Global Communications Conference (GLOBECOM)*. doi:10.1109/GLOCOM.2016.7841543

Glowacka, J., Krygier, J., & Amanowicz, M. (2015). A trust-based situation awareness system for military applications of the internet of things. 2015. *IEEE 2nd World Forum on the Internet of Things (WF-IoT)*. doi:10.1109/wf-iot.2015.7389103

Goel, A. K., Rose, A., Gaur, J., & Bhushan, B. (2019). Attacks, Countermeasures and Security Paradigms in IoT. *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*. doi:10.1109/icicict46008.2019.8993338

Jaitly, S., Malhotra, H., & Bhushan, B. (2017). Security vulnerabilities and countermeasures against jamming attacks in Wireless Sensor Networks: A survey. *International Conference on Computer, Communications, and Electronics (Comptelix).* doi:10.1109/COMPTELIX.2017.8004033

Kamble, A., Malemath, V. S., & Patil, D. (2017). Security attacks and secure routing protocols in RPL-based Internet of Things: Survey. *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*. doi:10.1109/ETIICT.2017.7977006

Khan, Z. A., Ullrich, J., Voyiatzis, A. G., & Herrmann, P. (2017). A Trust-based Resilient Routing Mechanism for the Internet of Things. *Proceedings of the 12th International Conference on Availability, Reliability and Security - ARES '17.* doi:10.1145/3098954.3098963

Kott, A., Swami, A., & West, B. J. (2016). The Internet of Battle Things. *Computer*, *49*(12), 70–75. doi:10.1109/MC.2016.355

Lahbib, A., Toumi, K., Elleuch, S., Laouiti, A., & Martin, S. (2017). Link reliable and trust aware RPL routing protocol for Internet of Things. *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*. doi:10.1109/nca.2017.8171360

López-Chau, A., Cervantes, J., López-García, L., & Lamont, F. G. (2013). Fisher's decision tree. *Expert Systems with Applications*, *40*(16), 6283–6291. doi:10.1016/j.eswa.2013.05.044

Mabodi, K., Yusefi, M., Zandiyan, S., Irankhah, L., & Fotohi, R. (2020). Multi-level trust-based intelligence schema for securing of internet of things (IoT) against security threats using cryptographic authentication. *The Journal of Supercomputing*, *76*(9), 7081–7106. Advance online publication. doi:10.1007/s11227-019-03137-5

Mathur, A., Newe, T., & Rao, M. (2016). Defence against Black Hole and Selective Forwarding Attacks for Medical WSNs in the IoT. *Sensors (Basel)*, *16*(1), 118. doi:10.3390/s16010118 PMID:26797620

Mehta, R., & Parmar, M. M. (2018). Trust based mechanism for Securing IoT Routing Protocol RPL against Wormhole &Grayhole Attacks. *2018 3rd International Conference for Convergence in Technology (I2CT)*. doi:10.1109/i2ct.2018.8529426

Perrey, H., Landsmann, M., Ugus, O., Wahlisch, M., & Schmidt, T. C. (2016). TRAIL: Topology authentication in RPL. *Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks, EWSN '16*.

Samie, F., Bauer, L., & Henkel, J. (2016). IoT technologies for embedded computing. *Proceedings of the Eleventh IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis - CODES '16*. doi:10.1145/2968456.2974004

Sankar & Srinivasan. (2018). Fuzzy Logic Based Energy Aware Routing Protocol for Internet of Things. *I. J. Intelligent Systems and Applications, 10*, 11-19. DOI: 10.5815/ijisa.2018.10.02

Sheng, Z., Mahapatra, C., Zhu, C., & Leung, V. C. M. (2015). Recent Advances in Industrial Wireless Sensor Networks Toward Efficient Management in IoT. *IEEE Access: Practical Innovations, Open Solutions*, *3*, 622–637. doi:10.1109/ACCESS.2015.2435000

Sung, W.-T., & Chiang, Y.-C. (2012). Improved Particle Swarm Optimization Algorithm for Android Medical Care IOT using Modified Parameters. *Journal of Medical Systems*, *36*(6), 3755–3763. doi:10.1007/s10916-012-9848-9 PMID:22492176

Varshney, T., Sharma, N., Kaushik, I., & Bhushan, B. (2019). Architectural Model of Security Threats & their countermeasures in IoT. *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*. doi:10.1109/ICCCIS48478.2019.8974544

Verma, A., & Ranga, V. (2019). ELNIDS: Ensemble Learning based Network Intrusion Detection System for RPL based Internet of Things. *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*. doi:10.1109/iot-siu.2019.8777504

Wang, B., Chen, X., & Chang, W. (2014). A light-weight trust-based QoS routing algorithm for ad hoc networks. *Pervasive and Mobile Computing*, *13*, 164–180. doi:10.1016/j.pmcj.2013.06.004

Wang, J., Yang, Q., & Ren, D. (2009). An Intrusion Detection Algorithm Based on Decision Tree Technology. *2009 Asia-Pacific Conference on Information Processing*. doi:10.1109/APCIP.2009.218

Winter, T. (2012). *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*. https://tools.ietf.org/html/rfc6550

Yuan, Z., & Wang, C. (2016). An improved network traffic classification algorithm based on Hadoop decision tree. *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*. doi:10.1109/ICOACS.2016.7563047

Zhang, T., Ji, X., & Xu, W. (2019). Cuckoo-RPL: Cuckoo Filter based RPL for Defending AMI Network from Blackhole Attacks. *2019 Chinese Control Conference (CCC)*. doi:10.23919/ChiCC.2019.8866139

Zhong, X., Lu, R., Li, L., & Zhang, S. (2017). ETOR: Energy and Trust Aware Opportunistic Routing in Cognitive Radio Social Internet of Things. *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. doi:10.1109/glocom.2017.8254213

K. Prathapchandran is an Assistant Professor in the Department of Computer Applications, Karpagam Academy of Higher Education (Deemed to be University), Coimbatore, Tamilnadu, India. He received his B.C.A degree in Computer Applications from Madurai Kamaraj University, India in the year 2005, the M.C.A degree from Gandhigram Rural Institute – Deemed University, India in the year 2008 and the M.Phil degree in Computer Science from Bharathidasan University, India in the year 2010. He received his Ph.D. degree in Network Security from Gandhigram Rural Institute -Deemed University, India in the year 2017. He has seven years of experience in teaching. He has more than twenty publications in reputed Journals and Conference proceedings and five publications as book chapters. Besides, he is a reviewer for various international journals. His areas of interest are Mobile Ad hoc Networks, Internet of Things, Trust Management in self-organized Networks. He is a life member of the Computer Society of India (CSI).

T. Janani is a research scholar in the Department of Computer Applications, Karpagam Academy of Higher Education (Deemed to be University), Coimbatore, Tamilnadu, India. She received his B.Sc Computer Science degree in from Madurai Kamaraj University, Madurai, India in the year 2011, the M.C.A degree from Gandhigram Rural Institute – Deemed University, Gandhigram, India in the year 2014 and the M.Phil degree in Computer Science from Bharath University, Chennai, India in the year 2016.