CHAPTER 4

TRUST-BASED RPL SECURITY SOLUTIONS AGAINST SELECTIVE FORWARDING ATTACK OVER IOT

This chapter proposes trust based secure RPL routing protocols named as TSF-RPL and MLT-IoT against selective forwarding attack. It also provides the system model and proposed methodology of such routing models further. It explains the protocol processes of the proposed approach in detail. Consequently, it describes the simulation setup and performance metrics of TSF-RPL and MLT-IoT. Finally, the simulation results of such routing protocols are depicted with suitable descriptions.

4.1 Selective Forwarding Attack and its Effects on RPL Routing

Firstly, the selective forwarding Attack is defined in (Karlof and Wagner, 2003). These malicious attackers aim in dropping the data traffic only, whereas correctly forwarding the control packets (Wazir et al., 2011). The selective forwarding attack is launched in various forms. In the first type, the malicious devices attempt to drop the data traffic generated by single or multiple nodes. Such behavior is modeled as a denial of service attacks in the network. Secondly, the selective forwarding attacker is launching an attack like a Black hole attack in which the malicious node tries to refuse the packets and also forwards the refused data via an indictable routing path. The main intention of a black hole type selective forwarder is to create unfaithful information about routing in the network. Thirdly, the subverted device tries to neglect some messages during transmission. This type of selective forwarding attack is launched at a lower level routing protocol in which the malicious node drops the packets frequently and it sends a fake acknowledgment to the sender node. As a final step, the selective forwarding attackers intentionally delay the data packets and also try to make the packets invalid in the network. Moreover, the main aim of a selective forwarding attack is to reduce the network performance by dropping or delaying the data packets. In real-time IoT applications, the sensors

are the critical IoT devices and that are most vulnerable to selective forwarding attacks. Generally, the IoT devices are randomly disseminated in the area of the network. In RPL, the nodes select a parent node based on a rank value. Initially, the selective forwarding attacker correctly forwards the DODAG Information Object (DIO) packets to attain a minimum less rank value. Thus, the attacker is selected as a parent node in the network. After that, the attacker drops the packets received from child nodes and launches the attack in the network, resulting in reduced network performance. Hence, it is crucial for identifying selective forwarding attacker nodes as it improves the network performance. This chapter proposes two trust based selective forwarding attack detection mechanisms for maximizing the routing efficiency over IoT.

4.2 Trust based Selective Forwarding Attack Detection in RPL (TSF-RPL)

TSF-RPL_is a trust-based secure RPL routing protocol, which detects selective forwarding attacks in IoT networks. For identifying the attacker nodes, TSF-RPL exploits two mechanisms such as trust evaluation and trust-based secure data forwarding. In the trust evaluation phase, TSF-RPL evaluates the trust value of nodes considering the routing behavior of each node. Initially, each node observes the behavior of neighboring nodes and measures its packet dropping rate. Then, the measured packet dropping rate is updated in the routing table. In the trust-based secure data forwarding phase, the TSF-RPL allows the nodes to determine the malicious behavior by comparing the packet drop rate with the fixed threshold value. If any malicious behavior is detected, the corresponding node sends a false alarm about the suspected node to the internet gateway node. Further, the gateway node estimates the trust value for the corresponding node for detecting the packet loss due to attack behavior or network conditions. Moreover, TSF-RPL successfully detects the attack and improves network efficiency.

4.2.1 System Model of TSF-RPL

Numerous smart devices are required to design the IoT system. The IoT comprises heterogeneous devices like smartphones, sensor nodes, and actuators, and each device has various software requirements and sensors. The device identity is denoted as ID. The IoT comprises n number of devices, and the initial trust value of such devices is equal to one. The TSF_RPL evaluates the trust value over a particular period T={ti, t2, …..tn}. After a particular t

time, the TSF-RPL assigns some devices and attackers and detects such attackers based on the trust value. The selective forwarding attacker drops the DIO messages of child nodes for injecting malicious behavior. The TSF-RPL fixes a threshold (Th) for detecting the attackers. Furthermore, the TSF-RPL determines the malicious parent node and also selects another secure parent node to route the information.

4.2.2 Trust Evaluation of TSF-RPL

The TSF-RPL mechanism determines the trust value of parent nodes based on the packet dropping behavior. During trust evaluation, TSF-RPL cogent the nodes to maintain a trust-based neighbor list over a particular time. Each node measures the trust of its neighborhood and updates the trust value in its table. The TSF-RPL also fixes a trust threshold value to detect the attack behavior. By comparing the current trust value with the threshold, the TSF-RPL detects the malicious nodes and it selects a secure routing path for data forwarding. Moreover, the trust-based security of TSF-RPL enhances the routing efficiency of RPL over IoT.

4.2.2.1 Neighbor List Maintenance Based on Trust

In order to disrupt the routing process, the selective forwarding attackers attempt to maliciously drop the packets of child nodes instead of forwarding the packets. To detect and enhance the RPL routing performance, the TSF-RPL evaluates the trust value of nodes. Initially, the trust value of all the nodes is assigned as one. Then, the trust updating occurs based on the routing behaviour of nodes. Every child node observes the routing behaviour of the parent node and reduces the trust value.

At the same time, a child node that has successfully searched the attacker node delivered information to their neighbours and gateway. By applying this information, the trust-based neighbour list is maintained by the nodes.

4.2.2.2 Observation of Dropping Packet

The attacker nodes can camouflage under the background of network conditions and increases the false positive rate of conventional trust-based techniques. In such circumstances, randomly selecting the threshold value is not adequate to accurately identify the selective forwarding attackers. Thus, the proposed work observes the packet dropping rate and network condition aware threshold to detect the attackers in the network accurately. Every node x computes the ratio of the dropped packet, D_r for its neighboring nodes, as shown in figure (4.1). Neighboring nodes of x is denoted as Ne_x. The number of dropped packets $D_{r(i)}$ is equal to the difference between the total number of packets sent to the neighboring nodes Ne or its child node(s) i Є x and the total number of packets sent by a child node Ne towards the gateway.

$$D_r(j) = \sum_{i=1}^{|Ch|} Rec_{ij} - Rec_{jx} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4.1)$$

In the above equation (4.1), the node j is a neighbor of node x, and |Ch| denotes the total number of children for a node j. The term Rec_{jx} represents the total number of packets sent from node x. The trust value is estimated using the equation (4.2). If the D_r is less than the threshold, the trust value is minimized by 0.1. When the trust value is reduced below 0.5, the node is identified as an attacker and has to be isolated from the network.

$$\text{Trust value of x} = \begin{cases} \text{Previous Trust} - 0.1 & \text{if Dr} < threshold \\ \text{Previous Trust} & \text{Otherwise} \end{cases} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4.2)$$

The network collision is the main reason behind the normal packet loss. In the case of randomly deciding the threshold value, by mistake, this may happen that legitimate nodes may be selected as the attackers due to the packet dropping caused by the packet collision. This work considers the collision scenario to decide the threshold value. A large number of child nodes connected to the parent node tend to network collision and packet dropping. This work presents the child node based threshold estimation for RPL. The number of child nodes of a node x decides the link quality. If the number of the child nodes is high, the chance of packet collision and packet dropping at a particular parent node is high. This link quality is used to measure the threshold of

D_r. Moreover, and the TSF-RPL estimates the trust value of nodes using equation (4.3).

$$\text{Threshold} = 1 - (\text{Child Nodes (x)} / \text{Neighboring Nodes (x)}) \ldots\ldots\ldots\ldots\ldots\ldots (4.3)$$

4.2.3 Current Trust-Based Secure Data Forwarding

Every node maintains the neighbour list along with the observed trust values of the child nodes. Whenever the trust value is reduced below the value of 0.5, it is announced as a low trusted node. The reason behind this to consider the 0.5 threshold value is that at least half of the neighbouring nodes are at least benign. If the network has more than 50% malicious, the feedback is negative. Whenever the node receives the DIO message or data packets from low trusted nodes, it discards the packet. If a node has below 0.5 trusts, it is announced as an attacker. Moreover, the node is rejected from the DODAG structure. Then a trusted DODAG structure the data packets are transmitted. The trust-based defence system is explained in the following algorithm (4.1).

Child node do {
For every t period {
Observes the data forwarding;
Measures the D_r Value for a parent node;
Measures the threshold based on the connected children;
Creates a suspected node list;
If receive DIO message from a node Є suspected nodes
{
Drop the packet;
}
Else
Broadcast the DIO packet by increasing the rank value by one;
} }

Algorithm 4.1: Trust-Based Selective Forwarding Attack Detection in RPL

4.3 Performance Evaluation of TSF-RPL

The TSF-RPL utilizes Contiki for evaluating its performance. The Contiki is an open-source operating system specially designed for sensor networks. The Contiki software is designed at the Swedish Institute of Computer Science in 2004. This open-source software is a powerful simulating and communication methodology for the IoT microcontrollers among the available network simulation tools. It runs as a virtual machine over an operating system operated by a VMware player. So, it is efficient and highly portable for code backing up. Cooja is a network simulator employed in a Contiki operating system that supports cross-level simulation in the sensor network. It enables concurrent simulation in terms of a low level for sensor node hardware to the high level of node behavior. By employing this simulation environment, developers are able to achieve their applications run on large-scale networks and providing a precise tuning of emulated hardware. TSF-RPL mechanism exploits the Cooja simulator in the Contiki operating system for evaluating the performance of the proposed defense system. For evaluation, the TSF-RPL is compared with existing work (Zhang et al., 2015). In evaluation, only a portion of existing work is implemented (Zhang et al., 2015) for detecting the selective forwarding attacker.

4.3.1 Simulation Setup and Performance Metrics of TSF-RPL

For analyzing the effectiveness of proposed TSF-RPL, the performance of the TSF-RPL is evaluated under various node density scenarios. To achieve the purpose, numbers of nodes are varied from 30, 40, and 50.

Table 4.1: Simulation Parameters of TSF-RPL

Simulator	Cooja

Nodes Variation	31, 41, 51
Network Area	150m X 150m
Communication Range	50m
Data Transmission Interval	20 Sec
Data Packet Size	127 Bytes
Transport Layer Agent	UDP
Routing Protocol	TSF-RPL
MAC	802.15.4
Overall Simulation Time	5 minutes

The performance of TSF-RPL is estimated using several metrics that are detection accuracy, throughput, overhead, and power consumption.

Detection Accuracy: Detection accuracy is the ratio of successfully detected attackers to the total number of attackers.

Throughput: It is defined as the total number of data packets delivered in simulation time.

Overhead: It is defined as the number of control packets involved in the data transmission process.

Power Consumption: It is defined as the amount of power consumed to deliver the data packets from source to destination.

4.3.2 Simulation Results of TSF-RPL

The performance of the proposed TSF-RPL scheme is compared with the existing trust-based RPL network by taking performance metrics in the y-axis and the number of attackers on the x-axis.

Number of Attacker Nodes Vs. Detection Accuracy: Figure 4.1, Figure 4.2, and Figure 4.3 demonstrate the detection accuracy results of both TSF-RPL and Trust-based RPL under various

node density scenarios by varying the number of attackers from 1 to 4. The figures depict that the TSF-RPL attains the better routing performance compared to existing trust-based RPL. The reason is that the proposed TSF-RPL incorporates a useful trust evaluation model that fixes a trust threshold for successful attack detection. Also, the gateway nodes are responsible for confirming the malicious activity and thus, it improves the routing efficiency. Moreover, Figure 4.1, Figure 4.2, and Figure 4.3 show that the TSF-RPL maintains the attack detection accuracy level under all node density scenarios when compared to existing trust-based RPL.

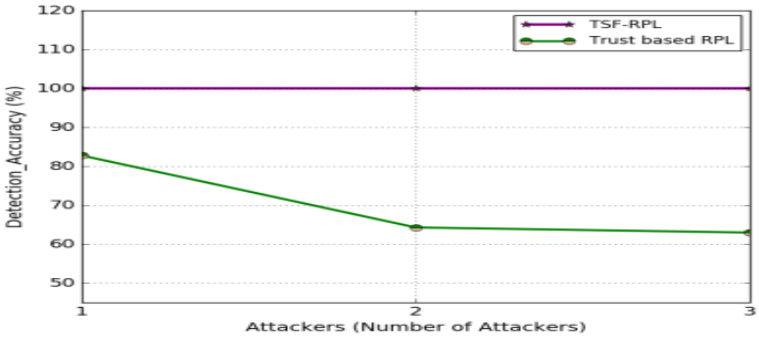

Figure 4.1: Number of Attacker Nodes Vs. Detection Accuracy for 31 Nodes

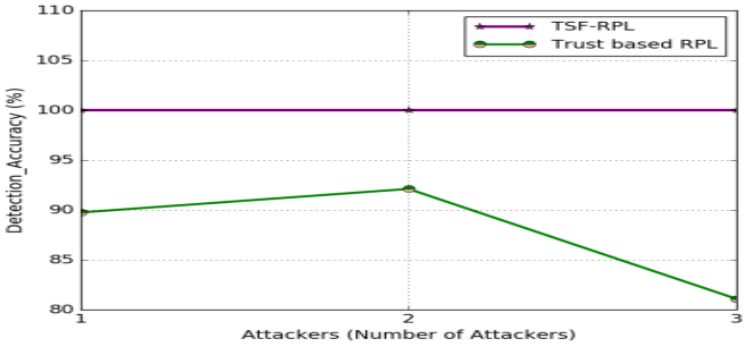

Figure 4.2: Number of Attacker Nodes Vs. Detection Accuracy for 41 Nodes

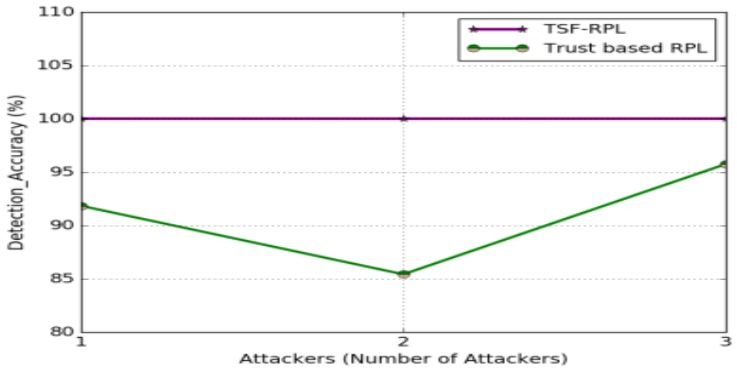

Figure 4.3: Number of Attacker Nodes Vs. Detection Accuracy for 51 Nodes

Number of Attacker Nodes Vs. Throughput: The throughput results of both TSF-RPL and trust-based RPL are comparatively evaluated in Figure 4.4, Figure 4.5, and Figure 4.6. The results are obtained by varying the number of attackers from 1 to 3 for node densities 30, 40, and 50. In figure 4.4, both protocols decrease the throughput by varying the number of attackers from low to high. However, the TSF-RPL attains high throughput results by 30% compared to trust-based RPL for the scenario of 3 attackers under the node density of 30. In Figure 4.5, the TSF-RPL has suddenly decreased the throughput from varying the number of attackers 1 to 2, whereas it slightly reduces the throughput after the point of 2 attackers. On the contrast, Figure 4.6 demonstrates that the TSF-RPL improves the throughput from 1 to 3 numbers of the attackers and reduces the throughput after point 4. Moreover, trust-based security in RPL improves network throughput considerably.

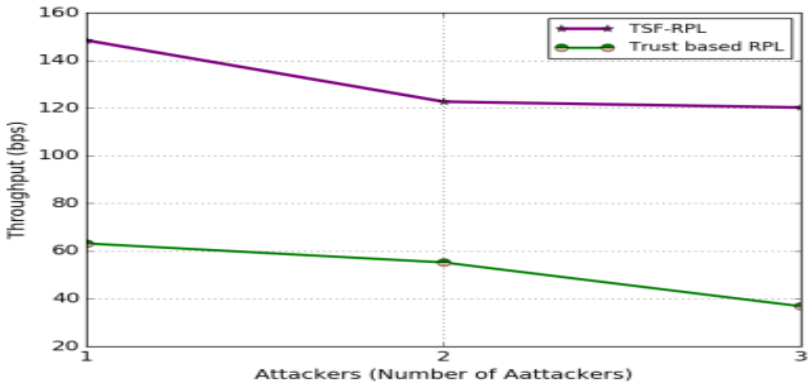

Figure 4.4: Number of Attacker Nodes Vs. Throughput for 31 Nodes

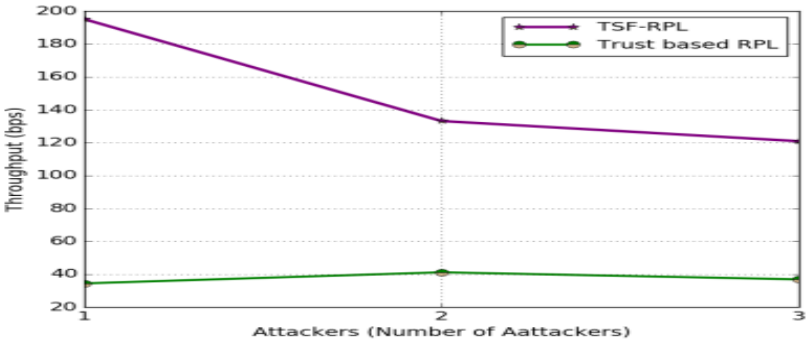

Figure 4.5: Number of Attacker Nodes Vs. Throughput for 41 Nodes

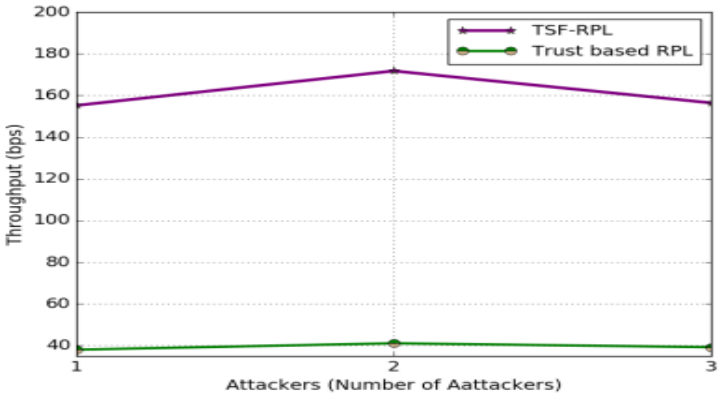

Figure 4.6: Number of Attacker Nodes Vs. Throughput for 51 Nodes

Number of Attacker Nodes Vs. Overhead: Figure 4.7, Figure 4.8, and Figure 4.9 portray the results of the overhead of both TSF-RPL and trust-based RPL under various node densities scenarios by increasing the node density from 1 to 4. From the results of Figure 4.7, Figure 4.8 and Figure 4.9, TSF-RPL reduces the overhead when compared to existing trust-based RPL. Both techniques exploit control packets for detecting the attackers. However, the total number of control packet transmission is significantly reduced in TSF-RPL compared to existing trust-based RPL even the numbers of attackers are increased in the network.

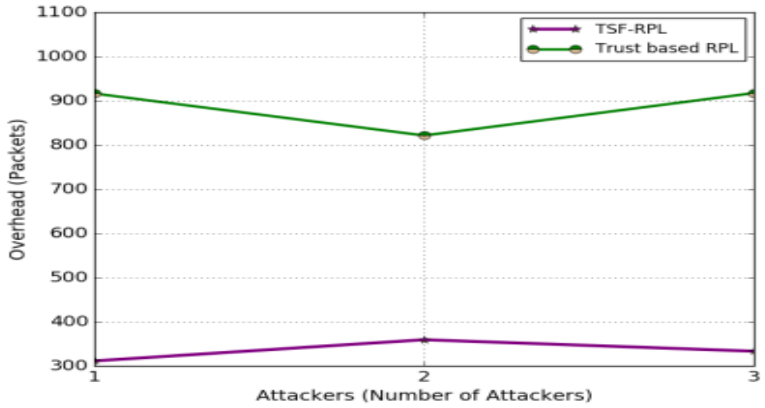

Figure 4.7: Number of Attacker Nodes Vs. Overhead for 31 Nodes

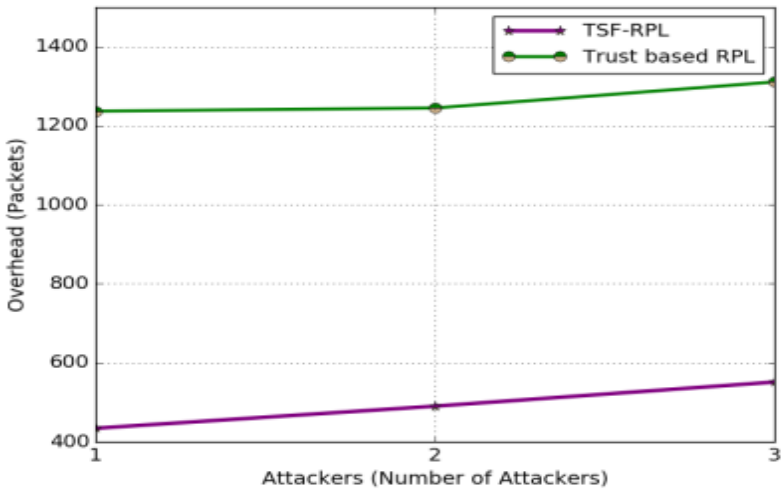

Figure 4.8: Number of Attacker Nodes Vs. Overhead for 41 Nodes

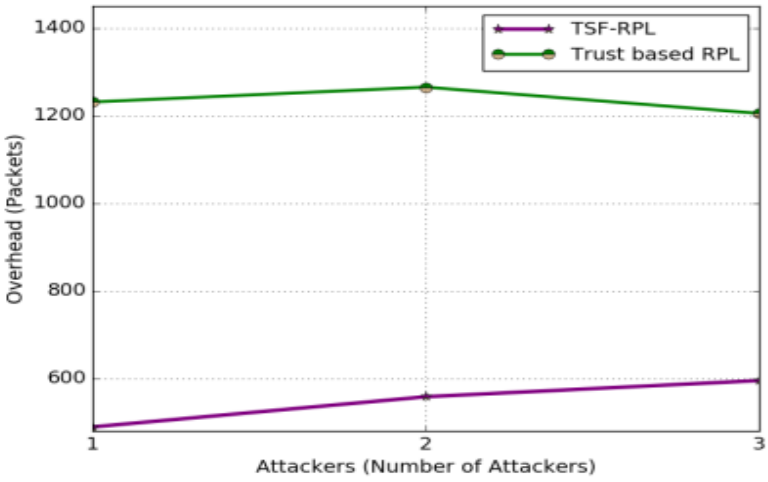

Figure 4.9: Number of Attacker Nodes Vs. Overhead for 51 Nodes

Number of Attacker Nodes Vs. Power Consumption

The comparative results of the power consumption of both TSF-RPL and trust-based RPL are shown in figure 4.10, Figure 4.11, and Figure 4.12. The IoT devices are generally battery-powered, and both techniques consumed some amount of energy for evaluating the trust level of nodes. Existing trust-based RPL initially determine parent and routing decisions with the help of RPL operation. Then trust was directly calculated based on successfully delivered, and the total sent nodes. However, the proposed TSF-RPL employs gateway nodes for confirming the attack behaviour. Thus, it reduces the power consumption level of nodes considerably, when compared with existing trust-based RPL. For instance, the proposed TSF-RPL improves the power consumption by 66.7%, when numbers of nodes are 30, and the numbers of attackers are 3 in the network.

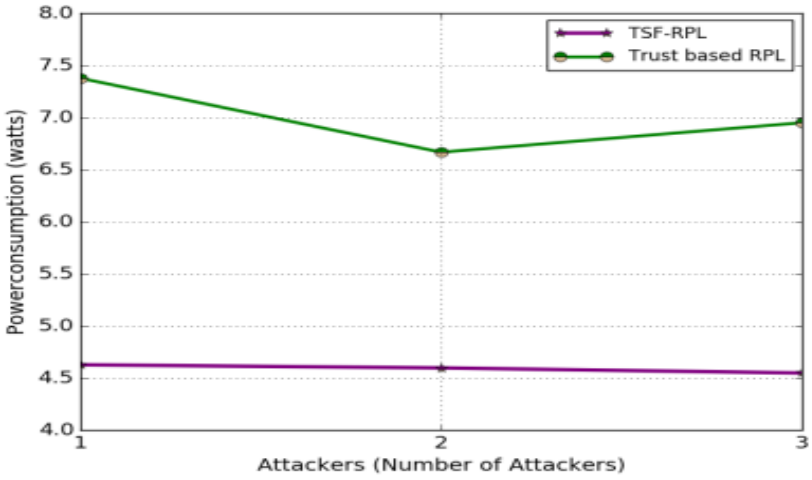

Figure 4.10: Number of Attacker Nodes Vs. Power Consumption for 31 Nodes

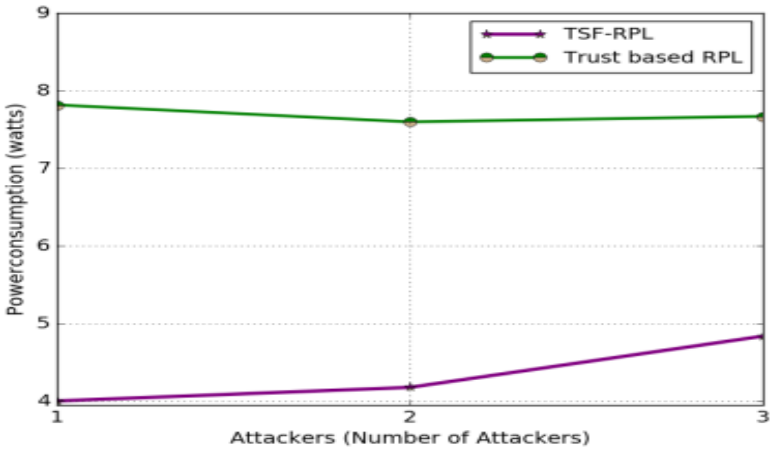

Figure 4.11: Number of Attacker Nodes Vs. Power Consumption for 41 Node

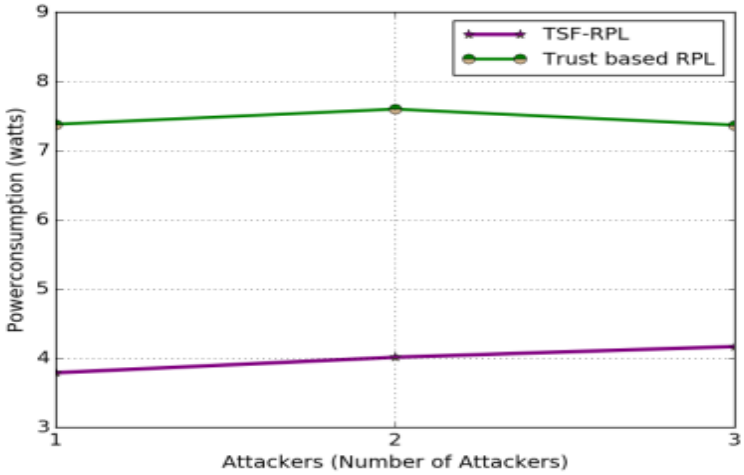

Figure 4.12: Number of Attacker Nodes Vs. Power Consumption for 51 Nodes

4.4 Multi-Level Trust-Based Secure RPL over IoT (MLT-IoT)

With MLT-IoT intends to detect the selective forwarding attack using a trust model that calculates trust in multi-level. At the initial stage, the devices in the network have a trust value equal to one. The selective forwarding attackers drop the received data packets in order to diminish the network efficiency. The multi-level trust model utilizes the concept of overhearing in a multi-level manner for detecting such attacks. In MLT-IoT, each node observes the

behaviour of its parent node in every transmission for estimating trust. The trust value is stored in the routing table over a particular time interval. If any suspicious behaviour is detected on the network, the nodes alert the gateway node using the trust information. Further, the gateway node involves in attack detection and confirmation process. Due to the nature of the dubious wireless environment, the trust evaluation may produce inaccurate results in MLT-IoT. In order to rectify such an issue, the MLT-it collects the second level trust information of a suspected node from the border nodes to confirm the malicious behaviour. Moreover, the proposed MLT-IoT enhances the IoT routing efficiency by the estimation of multi-level trust calculation.

4.4.1 Multi-Level Trust Evaluation

Primarily, the nodes in MLT-IoT have a trust value equal to one. After, the selective forwarding attacker aims to drop the received information of child nodes selectively and diminishes the data delivery efficiency in the network. In order to determine such attackers and improves the packet delivery rate, the child nodes in MLT-IoT observe the packet-forwarding behavior of their parent nodes by employing the overhearing method. The child nodes attach the estimated trust in the data packet for detecting malicious activities. For instance, a child node I send data packets to the parent node j and the trust relationship between i and j is computed using the following equation (4.4).

$$\mathrm{Trust}_{ij} = \mathrm{GDP}_i / \mathrm{FDP}_j \quad (4.4)$$

In equation (4.4), the term Trust_{ij} represents the trust relationship between nodes i and j. The terms GDPi and FDPj refer to the generated data packets of a node i and forwarded data packets by a node j respectively. Consequently, the MLT-IoT computes the trust based on parent opinion (Trust_{parent}) using equation (4.5). Consider, the node h is a parent node of j. Every parent node h computes the ratio of dropped packets at its child node j.

$$\mathrm{Trust}_{\mathrm{Parent(hj)}} = \mathrm{RDP}_j / \mathrm{RDP}_h \quad (4.5)$$

In equation (4.5), the node h is an observer, and node j is an observee. Due to the nature of the wireless environment, overhearing the packets may not accurate at all times, and it creates a significant impact on the trust estimation accuracy of MLT-IoT. To avoid such an issue, the MLT-IoT instructs the gateway nodes to calculate the second level of trust using the border nodes of a suspected node. Further, the gateway node receives the packet-forwarding behavior-based trust from the border nodes and estimates the final trust using the following equation (4.6). For instance, j is a suspected node, and the gateway evaluates the final trust value of node j, $Trust_{total(j)}$ as follows.

$$Trust_{total(j)} = \left(\sum_{X=1}^{Child\ nodes_j} Trust_{Child(xj)} + \sum_{y=1}^{Paren\ nodes_j} Trust_{Parent(yj)} \right) \Big/ x + y \ldots \ldots . (4.6)$$

Where the x is varied from 1 to the total number of child nodes of j (Child Nodes j), and the y value is varied from 1 to the total number of parent nodes of j (Parent Nodes j). Further, the MLT-IoT fixes a threshold to final the trust for detecting malicious activities in the network.

4.4.2 Selective Forwarding Attack Detection

After evaluating the trust of suspected nodes, the MLT-IoT performs the attack detection process using a fixed threshold value and updated multi-level trust information. If a final trust value of a node is less than 0.7, it is confirmed as a malicious node. Consequently, each node receives the multi fused level trust value of all the neighbouring nodes from the gateway. If the received trust value is minimum than the threshold, the nodes discard the data or DIO packets of the malicious nodes and start the DODAG structure formation. The MLT-IoT permits the nodes to exploit the final trust value as a rank in DODAG construction. Finally, the DODAG structure of MLT-IoT only comprises trustworthy nodes, and thus, it improves the routing efficiency and attack detection accuracy. Moreover, the malicious node information is disseminated in the network for neglecting the malicious activity.

4.5 Performance Evaluation of MLT-IoT

The effectiveness of MLT-IoT is evaluated using the Cooja simulator of the Contiki OS. For comparative evaluation, the existing Neighbor Based Trust Dissemination (NBTD) (Sonar and Upadhyay, 2016) is exploited in MLT-IoT implementation. The comparative evaluation results demonstrate that the proposed MLT-IoT attains better performance than NBTD in terms of attack detection accuracy and throughput.

4.5.1 Simulation Setup and Performance Metrics of MLT-IoT

To analyze the performance of MLT-IoT under diverse scenarios, the node density varies from 30 to 50, and the numbers of attackers vary from 1 to 5. The performance of MLT-IoT is estimated using various metrics such as detection accuracy, overhead, power consumption, energy consumption, and throughput.

Table 4.2: Simulation Parameters of MLT-IoT

Simulator	COOJA
Number of Nodes	31,41,51
Network Area	150m X 150m
Communication Range	100 m
Data Transmission Interval	20 sec
Data Packet Size	127 bytes
Transport Layer Agent	UDP
Routing Protocol	MLT-IoT
MAC Layer Protocol	802.15.4
Simulation Time	5 minutes

4.5.2 Simulation Results of MLT-IoT

Figure 4.13, Figure 4.14, and Figure 4.15 demonstrate that the detection accuracy of MLT-IoT and NBTD by varying the number of attackers from low to high. The detection accuracy of

MLT-IoT is higher than the existing NBTD. The estimation of multi-level trust using child and parent nodes in MLT-IoT enhances the detection accuracy. The nodes also receive the final trust value from the gateway node. The gateway calculates final trust value based on dropping rate with the involvement of all children and available neighbors in place of advice taken from a single node. The MLT-IoT also fixes a threshold based on the reports collected from neighbors, and it improves the accuracy of attack detection. Due to this reason, the detection accuracy of MLT-IoT is approximately 100%, as demonstrated in Figure 4.13, Figure 4.14, and Figure 4.15

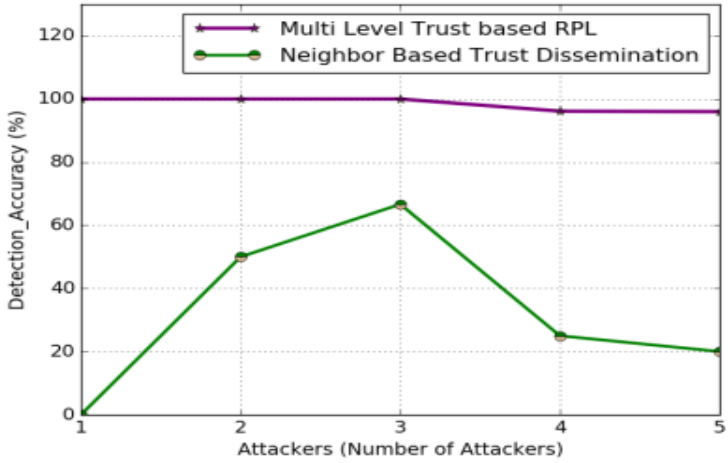

Figure 4.13: Number of Attacker Nodes Vs. Detection Accuracy for 31 Nodes

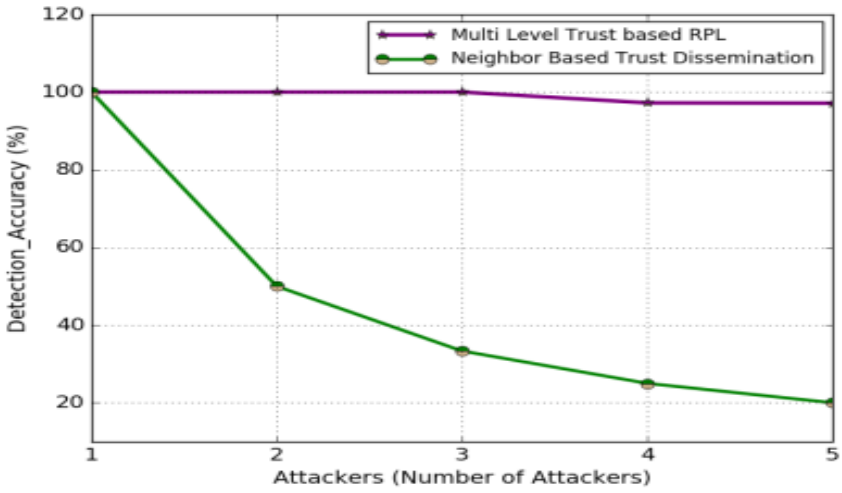

Figure 4.14: Number of Attacker Nodes Vs. Detection Accuracy for 41 Nodes

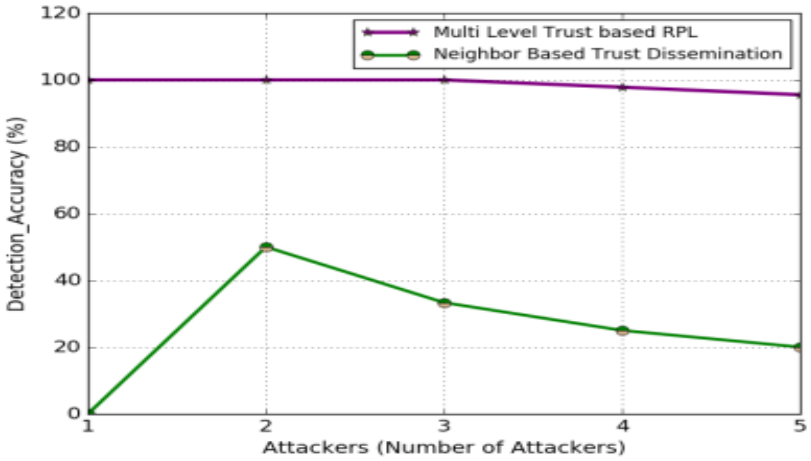

Figure 4.15: Number of Attacker Nodes Vs. Detection Accuracy for 51 Nodes

Number of Attacker Nodes Vs. Overhead: Figure 4.16, Figure 4.17, and Figure 4.18ss portray the overhead results of both MLT-IoT and NBTD mechanism by varying the number of attackers from 1 to 5. Figure 4.16, Figure 4.17, and Figure 4.18 illustrate the overhead results obtained for node densities 30, 40, and 50, respectively. The MLT-IoT employs an adequate amount of control messages for trust evaluation and attack detection over IoT networks.

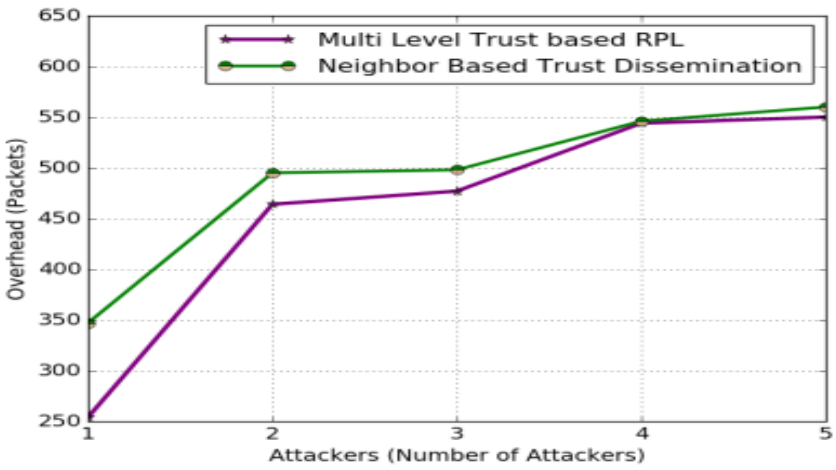

Figure 4.16: Number of Attacker Nodes Vs. Overhead for 31 Nodes

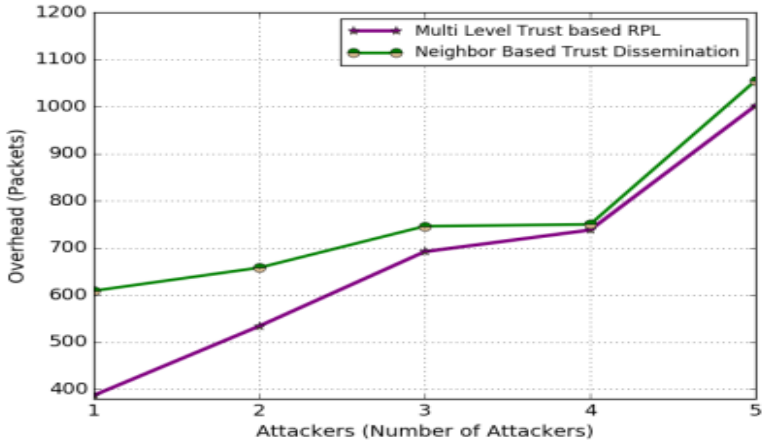

Figure 4.17: Number of Attacker Nodes Vs. Overhead for 41 Nodes

Thus, it increases the overhead of MLT-IoT, when high numbers of attackers are present in the network. However, the total numbers of control packets of MLT-IoT are significantly reduced compared to existing NBTD, even a high number of attackers present in the network.

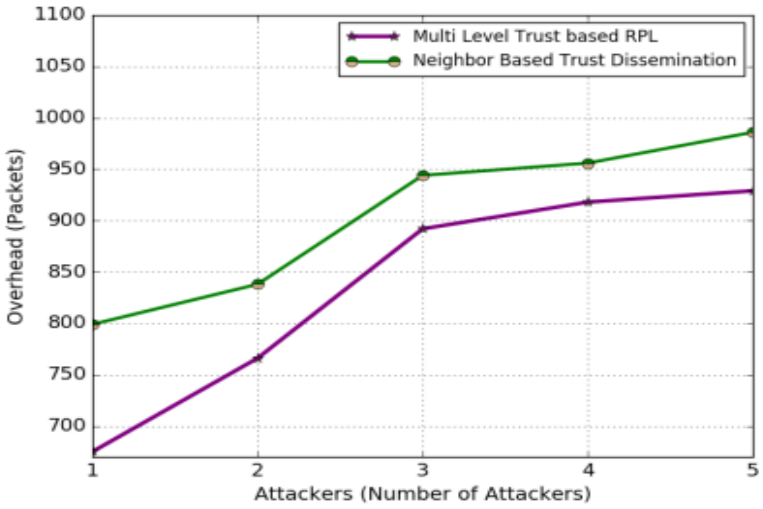

Figure 4.18: Number of Attacker Nodes Vs. Overhead for 51 Nodes

Number of Attacker Nodes Vs. Power Consumption: Figure 4.19, Figure 4.20, and Figure 4.21 illustrate the power consumption results that are comparatively obtained for MLT-IoT and NBTD by varying the number of attackers are varied from 1 to 5. Further, the results obtained

for various node densities 30, 40, and 50 are demonstrated in Figure 4.19, Figure 4.20, Figure 4.21 respectively.

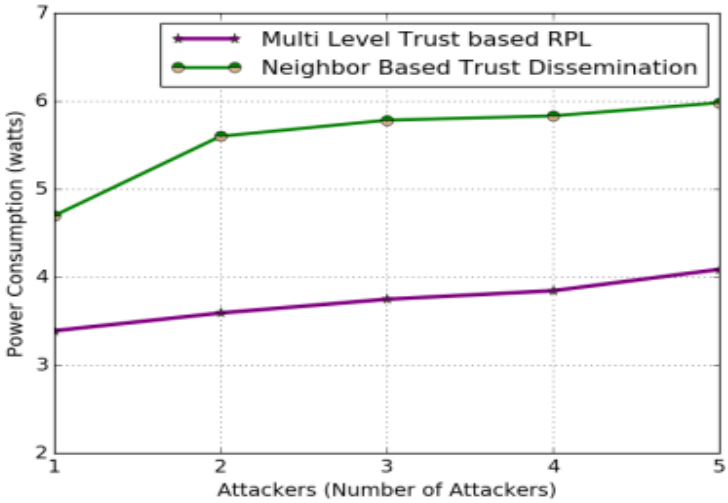

Figure 4.19: Number of Attacker Nodes Vs. Power Consumption for 31 Nodes

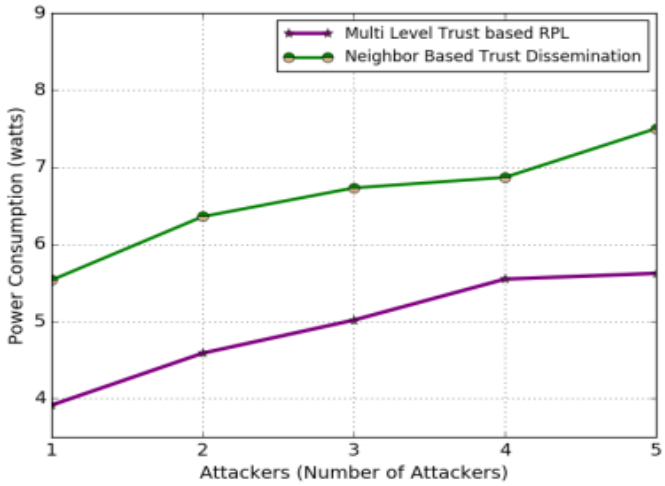

Figure 4.20: Number of Attacker Nodes Vs. Power Consumption for 41 Nodes

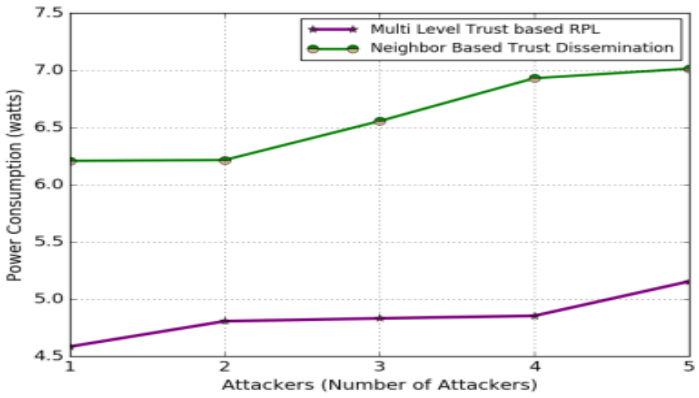

Figure 4.21: Number of Attacker Nodes Vs. Power Consumption for 51 Nodes

Both MLT-IoT and NBTD escalate the power consumption by increasing the number of attackers from low to high. However, the power consumption of MLT-IoT is minimum compared with the existing NBTD mechanism. The reason is that the MLT-IoT utilizes the gateway node for trust evaluation, resulting in reduced power consumption at the IoT nodes.

Number of Attacker Nodes Vs. Energy Consumption: The energy consumption results of both MLT-IoT and NBTD are demonstrated in Figure 4.22, Figure 4.23, and Figure 4.24 by varying the number of attackers from low to high.

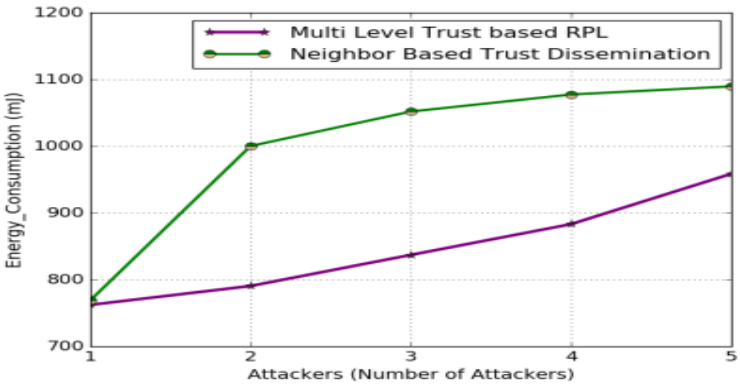

Figure 4.22: Number of Attacker Nodes Vs. Energy Consumption for 31 Nodes

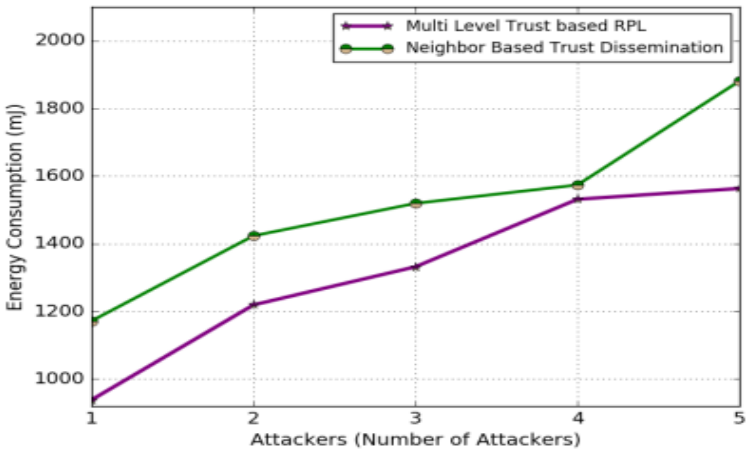

Figure 4.23: Number of Attacker Nodes Vs. Energy Consumption for 41 Nodes

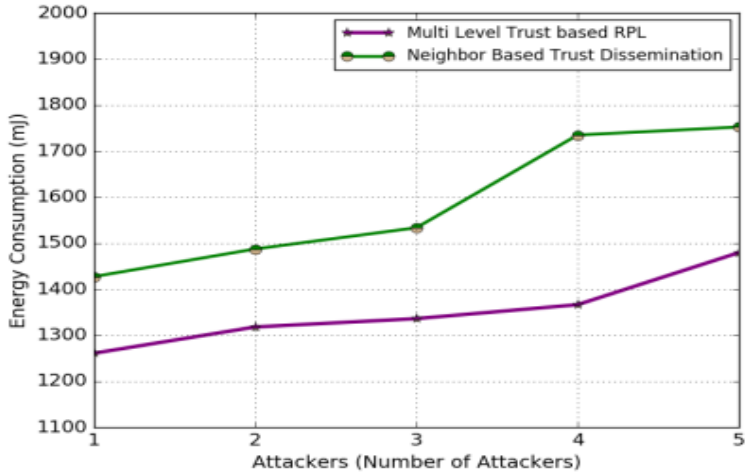

Figure 4.24: Number of Attacker Nodes Vs. Energy Consumption for 51 Nodes

To evaluate the effectiveness of MLT-IOT, the results are varied in various numbers of nodes like 30, 40, and 50. From the results of Figure 4.22, Figure 4.23 and Figure 4.24, both systems increase energy consumption by varying the number of attackers from 1 to 5. However, the proposed MLT-IoT employs the gateway nodes for detecting the attackers, and thus, it reduces the energy consumption level in the network.

Number of Attacker Nodes Vs. Throughput: Figure 4.25, Figure 4.26, and Figure 4.27 show the performance results of the throughput of both MLT-IoT and NBTD mechanisms. The throughput performance is estimated by varying the number of attackers and the number of nodes. In both mechanisms, the throughput is high under low attack scenarios, and it is reduced in high attack scenarios.

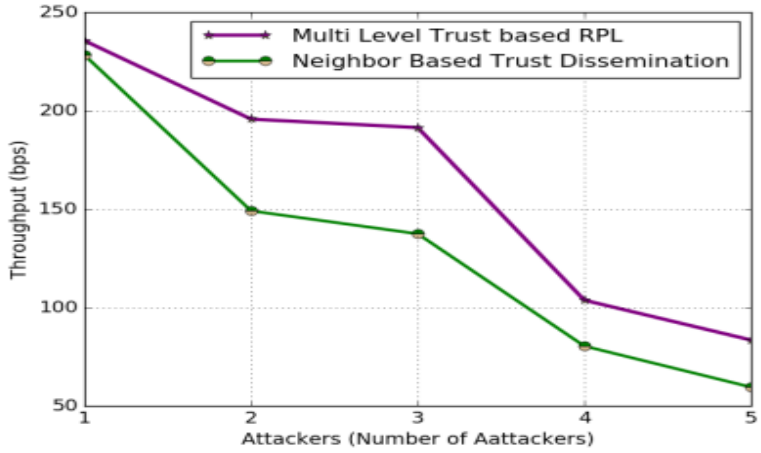

Figure 4.25: Number of Attacker Nodes Vs. Throughput for 31 Nodes

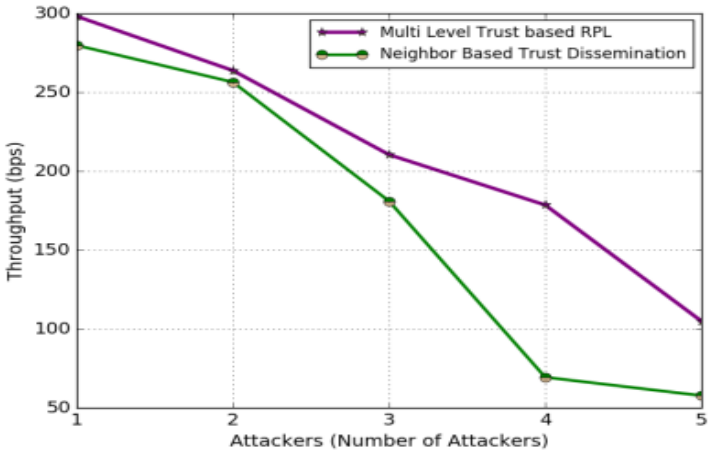

Figure 4.26: Number of Attacker Nodes Vs. Throughput for 41 Nodes

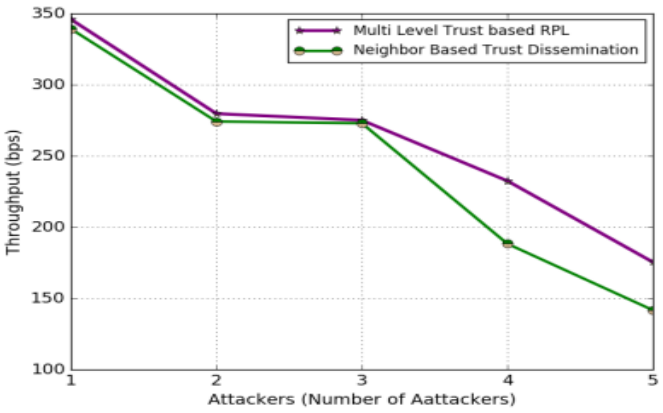

Figure 4.27: Number of Attacker Nodes Vs. Throughput for 51 Nodes

The MLT-IoT and NBTD attain nearly equal performance when low numbers of attackers are presented in the network. Nevertheless, the MLT-IoT remarkably performs than NBTD under high attack scenarios.

SUMMARY

Chapter 4 discussed the proposed TSF-RPL mechanism and MLT-IoT mechanism in detail. Initially, the effect of a selective forwarding attack in RPL is explained. In the TSF-RPL mechanism, two phases, such as trust evaluation of nodes based on routing behavior and trust-based secure forwarding, are explained in detail. The system model of the proposed methodology is discussed. Then, the performance analysis of TSF-RPL and existing trust-based RPL is performed using performance metrics such as detection accuracy, throughput, overhead, power consumption. The MLT-IoT mechanism is explained along with the trust evaluation procedure and selective forwarding attack detection using a multi-level trust mechanism. The performance results of the proposed MLT-IoT model in comparison with the NBTD mechanism using performance metrics such as detection accuracy, overhead, energy consumption, power consumption, and throughput are presented.