**TRUSTWORTHY MANET ROUTING ESTAODV IMPLEMENTATION USING DEEP REINFORCEMENT LEARNING**

Sajith Chamara Gunawardene Liyanage

IT14098888

Degree of Bachelor of Science Special (Honors) in Information Technology

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

October 2018

**TRUSTWORTHY MANET ROUTING ESTAODV IMPLEMENTATION USING DEEP REINFORCEMENT LEARNING**

Sajith Chamara Gunawardene Liyanage

IT14098888

Dissertation submitted in partial fulfillment of the requirements for the degree of Bachelor of Science Special (Honors) in Information Technology

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

October 2018

# DECLARATION

“I declare that this is my own work and this dissertation” does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to Sri Lanka Institute of Information Technology the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other media. I retain the right to use this content in whole or part in future works (such as articles or books).

**Author:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Registration No | Signature | Date |
| Liyanage S.C.G. | IT14098888 |  |  |

The above candidate has carried out research for the B.Sc. Dissertation under my supervision.

**Supervisor:**

|  |  |  |
| --- | --- | --- |
| Name | Signature | Date |
| Mr. Prabath Lakmal Rupasinghe |  |  |

# Abstract

A collection of nodes which have the ability to move randomly within a wireless network is called a mobile ad hoc network (MANET). MANET plays a major role in wireless communication technology. Data transferring within the network has two considerable facts, reliability and security. Ensuring security in a mobile ad hoc network is a major concern due to the unpredictable motions and behaviors of network nodes.

In a wireless mobile network, it is possible for a large number of data packets to transmit among nodes within a small period of time. Therefore, it is possible that some nodes might not behave as we expect. It can eventually cause to a considerable amount of data packet drops. It shows that the existing security mechanisms have failed to distinguish between trustworthy and malicious nodes. In order to further categorize malicious nodes, the spiral model has introduced. It is capable of distinguishing pure malicious and collaborative malicious nodes. Usually, the nodes select the shortest path; but sometimes it may not be the reliable route to transfer data. Therefore, Reinforcement Learning (RL) component has proposed to predict the trustworthy routes.

Keywords— MANET, Spiral model, RL component

# ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to our project supervisor, Mr. Prabath Lakmal Rupasinghe for his patient guidance, and useful critique of our research work. And I would be grateful to the corporation given by Ms. Chamira Nawarathna to make the research project for a successful one. Their willingness to give their time so generously has been very much valuable. Especially mentioning the moral support and the continuous guidance by providing important feedback enabled us to complete our work successfully. I would also like to thank our lecturer-in-charge, Mr. Jayantha Amararachchi for constantly providing us with the guidance and assistance to carry out the research successfully. Also, I would like to thank my parents and friends who supported and encouraged me throughout the research project.

# TABLE OF CONTENTS

[DECLARATION iii](#_Toc526932315)

[Abstract iv](#_Toc526932316)

[ACKNOWLEDGEMENT v](#_Toc526932317)

[TABLE OF CONTENTS vi](#_Toc526932318)

[LIST OF TABLES vii](#_Toc526932319)

[LIST OF FIGURES vii](#_Toc526932320)

[LIST OF ABBREVIATIONS viii](#_Toc526932321)

[1. INTRODUCTION 1](#_Toc526932322)

[**1.1.** **Introduction** 1](#_Toc526932323)

[**1.2.** **Literature Survey** 2](#_Toc526932324)

[**1.2.1.** **Authentication using the trust to detect misbehaving nodes in mobile ad-hoc networks using Q-Learning [1]** 2](#_Toc526932325)

[**1.2.2.** **Information theoretic framework of trust modeling and evaluation for ad hoc networks [2]** 3](#_Toc526932326)

[**1.2.3.** **Different ways to achieve trust in MANET [3]** 4](#_Toc526932327)

[**1.2.4.** **Secure routing with AODV protocol for mobile ad hoc networks [4]** 5](#_Toc526932328)

[**1.2.5.** **QoS assertion in manet routing based on trusted AODV (ST-AODV) [5]** 5](#_Toc526932329)

[**1.2.6.** **EBoX: Evidence of behavior information exchange mechanism against selfish attacks [6]** 6](#_Toc526932330)

[**1.2.7.** **QoS of MANET through trust-based AODV routing protocol by exclusion of black hole attack [7]** 6](#_Toc526932331)

[**1.2.8.** **Black hole attack in AODV routing protocol [1]** 7](#_Toc526932332)

[**1.3.** **Research Problem and Research Gap** 7](#_Toc526932333)

[**1.4.** **Research Objectives** 9](#_Toc526932334)

[2. METHODOLOGY 10](#_Toc526932335)

[**2.1.** **Spiral Model** 10](#_Toc526932336)

[**2.1.1.** **Collaborative malicious node discovery process** 10](#_Toc526932337)

[**2.1.2.** **Penalty phase** 12](#_Toc526932338)

[**2.2.** **Deep Reinforcement Learning Model** 13](#_Toc526932339)

[**2.3.** **Implementation** 15](#_Toc526932340)

[**2.3.1.** **Technologies** 15](#_Toc526932341)

[3. RESULTS AND DISCUSSION 16](#_Toc526932342)

[**3.1.** **Spiral Model** 16](#_Toc526932343)

[**3.2.** **Deep Reinforcement Learning Model** 18](#_Toc526932344)

[4. CONCLUSION 20](#_Toc526932345)

[REFERENCES 21](#_Toc526932346)

# LIST OF TABLES

[Table 1.1: Backup Table 10](#_Toc526685038)

# LIST OF FIGURES

[Figure 1.1: Sample Network Diagram with 3 Network Nodes 3](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597002)

[Figure 2.1: Flow Chart for the Spiral Model 11](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597003)

[Figure 2.2: System Diagram 14](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

[Figure 3.1: Sample Network Topology 16](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

[Figure 3.2: Backup Table Data for Node 1 17](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597004)

[Figure 3.3: Backup Table Data for Node 2 17](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

[Figure 3.4: Backup Table Data for Node 3 17](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

[Figure 3.5: Backup Table Data for Node 4 17](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

[Figure 3.6: Backup Table Data for Node 5 17](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597004)

[Figure 3.7: Q-values for Flows in Node 1 18](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597004)

[Figure 3.8: Q-values for Flows in Node 2 18](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597004)

[Figure 3.9: Q-values for Flows in Node 3 19](file:///D:\SLIIT\4th%20year\Research\DocRepo\Doc\18-024_IT14098888_Thesis.docx#_Toc526597004)

[Figure 4.0: Q-values for Flows in Node 4 19](file:///D:\\SLIIT\\4th%20year\\Research\\DocRepo\\Doc\\18-024_IT14098888_Thesis.docx" \l "_Toc526597004)

s

# LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| ABED | Ant-Based Evidence Distribution |
| AODV | Ad hoc On Demand Distance Vector |
| DQN | Deep Q Network |
| EBox | Evidence of Behavior Information Exchange |
| GRE | Generalized Reputation Evidence |
| IDM | Intrusion Detection Model |
| IRM | Intrusion Response Model |
| MANET | Mobile Ad hoc Network |
| NFC | Near Field Communication |
| PKI | Public Key Infrastructure |
| QoS | Quality of Service |
| REP | Recommendation Exchange Protocol |
| RREQ | Route Request packet |
| RL | Reinforcement Learning |
| TL | Trust Level |
| TRR | Trust Recommendation Request |

# INTRODUCTION

## **Introduction**

Wireless communication is a communication mode which does not use physical wires to connect between two or more devices to transfer data. It uses electromagnetic waves to transfer signals. Depending on the wave frequencies, network coverage area will be changed. It can occur network connectivity issues for some regions. Generally, there are more advantages of using wireless networks. Cost is low since it does not require any physical infrastructure to maintain. Most of the times flexibility and accessibility of a wireless network are high regardless of the location. Some of the popular wireless technologies are WiFi, Bluetooth, NFC (Near-field communication) and satellite services. Routing protocols specify how routers should communicate with each other in the network with aid of such technologies. In a mobile ad hoc network, the ad hoc routing protocol is used for this purpose.

Due to the mobility feature of network nodes in MANET, security issues could arise at any time. Simply the packets might be dropped due to some unpredictable conditions. Therefore, the regular transmission process of the network can be interrupted. Existing cryptographic techniques like public/private key encryption and other security mechanisms such as packet filters, firewalls cannot always identify the trustworthy nodes to communicate. In public/private key encryption, anyone can encrypt a message using the public key of the receiver. As diverse to all the above-mentioned methods, defining a trust-based schema on top of AODV to detect each one hop (directly connected) neighbor nodes was solved this issue up to a considerable level.

Trustworthiness of nodes in the ad-hoc network was evaluated by global trust value which was a combination of direct trust and indirect trust values. Direct trust was the trust which has built with the experience among directly connected nodes and when a node took recommendations regarding a particular node from other neighbor nodes, simply it could be considered as taking the indirect trust. Based on the global trust value nodes were categorized as trustworthy, partially trustworthy, selfish and malicious nodes. Their malicious category could further divide into pure malicious and collaborative malicious through the **Spiral model** which will be reviewed ahead in this documentation. Next step was to determine the best route path in the ad hoc network using **Reinforcement learning (RL) model.** Before step into that model since the system had already categorized network nodes as mentioned above, we could expect some performance wise efficient in the system.

## **Literature Survey**

### **Authentication using the trust to detect misbehaving nodes in mobile ad-hoc networks using Q-Learning [1]**

Authentication which was the key factor to be considered in MANET could be categorized into two sections called pre-authentication and post-authentication. As the name denotes pre-authentication was initial network deployment and post-authentication was a mechanism to detect nodes in the network over a period of time. According to S.Sivagurunathan, K.Prathapchandran, and A.Thirumavalavan, trust could be defined as “*the reliability, timeliness, and integrity of message delivery to a node’s intended next hop*” [1].

Nodes in an ad-hoc network eventually were categorized into three sections such as trustworthy, partially trusted and untrusted; based entirely on their direct trust. So, it was unwise to come to conclusions based only on their direct trust value. There could also exist indirect aspects throughout the network which might affect the trust between nodes. In that case, apart from the direct trust, an indirect trust value which could consider such indirect factors should be calculated. Afterward, a global trust value could be defined based on the average value of both direct and indirect trust values and that global trust could be used for a rewarding system within the network.

### **Information theoretic framework of trust modeling and evaluation for ad hoc networks [2]**

It was preferred to consider the recommendation values from other nodes to fulfill the requirement of calculating indirect trust. Yan Lindsay Sun, Wei Yu, Zhu Han, and K.J. Ray Liu had proposed an information theoretic framework as a solution. According to them, trust is a “*measure of uncertainty with its value represented by entropy”* [2].

This was a better approach than the 1.2.1 solution to detect misbehaviors of nodes because it defined a combination of two trust models named ‘entropy-based model’ and ‘probability-based model’. Under the entropy-based model, they came up with an equation to calculate TABC which was the same as the indirect trust between node A and C.



Figure 1.1: Sample Network Diagram with 3 Network Nodes

RAB was the recommendation value from node A to B and TBC was the trust value from node B to C. Probability-based model could calculate the multipath trust propagation and concatenation using probability equations. Probability values for the trust relationship could be converted into trust values using entropy-based equations. In order to calculate indirect trust, it was required to request recommendations from other nodes. A new control packet introduced as TRR (Trust Recommendation Request) to get the trust value of a particular node by requesting from the other neighbor nodes.

TABC = RABTBC

According to the Figure 1.1, if node A wanted to know the indirect trust value of node C, node A could send a TRR message to node B by requesting for the trust value of node C.  That trust value was only available in node B’s trust table. Finally, TABC could be evaluated as in the above equation. Based on that trust value they were attempting to detect malicious nodes.

One drawback of this solution was that malicious nodes could collaboratively provide wrong recommendations for other nodes. Therefore, a mechanism should be required to detect collaborative malicious nodes. By analyzing this past history of network node interactions, we came up with a solution to categorizing nodes into levels based on the global trust which could be utilized to identify the malicious nodes. Network nodes could be trustworthy, partially trustworthy, selfish, pure malicious or collaborative malicious nodes.

### **Different ways to achieve trust in MANET [3]**

It was important to study the existing trust calculation schemes. Each scheme had different unique features, merits, and findings. There were five main trust schemes in MANETs.

* Protocol based scheme - Basically, security protocols had been implemented in this scheme. ABED (Ant-Based Evidence Distribution) scheme utilized a swarm intelligence paradigm [8] to model the protocol-based schema. Communication among network nodes happened through agents similar to ants [3] in ABED. Ants collected information which was called as pheromones [3]. Based on pheromones, ants found an optimal path to measure trust evidence. Generalized Reputation Evidence (GRE) [3], was another instance for the protocol-based scheme.
* System level-based scheme - Under system level-based, it could give rewards to trustworthy nodes and gave a penalty to malicious or selfish nodes. Because of this purpose, they had defined some trust models. Watchdog trust model could detect selfish nodes and Collaborative Reputation trust scheme could distinguish selfish nodes and malicious nodes.
* Cluster based scheme
* Maturity based scheme
* PKI (Public Key Infrastructure) based scheme

They had not yet implemented a way to further categorize malicious nodes.

### **Secure routing with AODV protocol for mobile ad hoc networks [4]**

T. Farid and A. Prahladachar had basically defined two types of security attacks and two types of models as the proposed solution. Compromised network nodes and selfish network nodes could make internal attacks. When a network node did not send or forward data packets and become inactive when other nodes needed them and become active only for its own benefits, it could be named as a ‘selfish node’. There was also another type of attack called ‘external attack’ which was occurred due to invalid cryptographic information. Intrusion Detection Model (IDM) and Intrusion Response Model (IRM) came into the front as the solution. IDM used neighbor node information to detect misbehaving nodes. If misbehavior count was greater than the threshold value, it could broadcast about that misbehaving node to other nodes. Under IRM, if two or more network nodes report about the same node, Purge packet [4] was transmitted to isolate the malicious node from the network.

### **QoS assertion in manet routing based on trusted AODV (ST-AODV) [5]**

In order to increase Quality of Service (QoS) in ad hoc networks, Sridhar Subramanian and Baskaran Ramachandran had proposed a trusted AODV called “ST-AODV”. The trust level (TL) value was calculated as below.

TL = T(RREQ)\*Qr + T(RREP)\*Qp + T(DATA)\*Qd

There Qr, Qp, and Qd were the intermediate values to calculate the request rate, reply rate and data transmission rate of network nodes respectively. And time factor to evaluate the route request, route reply and data sent were measured via T(RREQ), T(RREP) and T(DATA) accordingly. For a given network node, if TL was less than or equaled to a threshold value, then it was considered as an untrustworthy node who might drop packets. Otherwise, it was a trustworthy node who should be allowed to stay in the network for a better-secured communication.

### **EBoX: Evidence of behavior information exchange mechanism against selfish attacks [6]**

Evidence of Behavior Information Exchange (EBox) mechanism could detect selfish nodes and misbehaving nodes in the ad hoc network and ignored such nodes from the network. It used a reputation-based schema to award penalty points for selfish network nodes and credit points for trustworthy network nodes. Santhosh J and Malini V K proposed this mechanism to estimate trust values by comparing their predefined threshold values.

### **QoS of MANET through trust-based AODV routing protocol by exclusion of black hole attack [7]**

Radha Krishna Bar, Jyotsna Kumar Mandal, and Moirangthem Marjit Singh had proposed to evaluate trust values for nodes in the ad hoc networks based upon two criteria which their ability to forward data packets and forward RREQ for a given network node. Finally, the trust value was calculated as a multiplication of the forwarded data packets ratio and the forwarded RREQ packets ratio. Then trust value was recorded in the routing table to make routing decisions.

### **Black hole attack in AODV routing protocol [1]**

One of the most popular attacks in AODV (Ad hoc On-Demand Distance Vector) routing, was the ‘Blackhole attack’. When the source needed to send packets to an unknown destination which was not included in sender’s routing table, it could send RREQ (route request) packets to neighbor nodes to identify the destination node. In that case, a black hole node could send the reply packet (RPLY) before all the other nodes, it could be chosen as the route to transfer data. Black hole node receiving the information in the packets and drops them without passing towards the destination. Since the sender did not suspect neighbor nodes in the existing network, it could continue to communicate with the harmful black hole node. Eventually, that lead to making the packet dropping ratio into a higher value.

Apart from RREQ and RPLY, there were a different number of other control packets are used in AODV protocol. RERR (route error) message was sent by neighbor nodes whenever a link failure happened or destination was unreachable. Basically, HELLO packets were broadcasted to identify the available neighbor nodes in the network. After a time period, we could count the number of forwarded and received data packets. Using that information, we could calculate the trust between directly connected nodes.

## **Research Problem and Research Gap**

Nodes in MANET could move randomly without any centralized structure or any time pattern. Due to this self-configuration and self-optimization characteristics, such networks could be called self-organized networks [1]. It was difficult to provide security for such dynamic environments than traditional networks. Ad hoc networks like MANET were vulnerable to various attacks due to these dynamic and distributed behaviors of nodes. This could lead to many IoT device failure with resource-constrained environments. Therefore, there should be mechanisms which allowed a node to measure the reliability and security of other nodes. Then trustworthy nodes could avoid dealing with malicious nodes. As a result, it could improve both network performance and security aspects.

As revealed in 1.2.1, only the direct trust was calculated to evaluate the trustworthiness of nodes. That could cause problems in capturing indirect behaviors of network nodes that brought harm. There was no way to prove complete trustworthiness only depended on direct interactions among each node in the network. There might have chances of getting high accuracy for trust values by getting recommendations from other network nodes. At the same time could not come to a better decision only depending on indirect trust value. That could raise the requirement of calculating the average value of direct trust value and indirect trust value when taking a better conclusion on the trustworthiness of nodes. On the other hand, a definition of trust among the network nodes was similar to trust among human beings. Direct trust was the trust which built with the experience among each other. When someone had suspected about that trust, going to take recommendations from others was the indirect trust. Therefore, measuring both direct trust and indirect trust was a vital factor.

According to 1.2.2, they did not consider the collaborative behaviors of malicious nodes. Sometimes a group of malicious nodes provided wrong recommendations to make a node in their team as more trustworthy. Eventually, it also contributed to a considerable amount of packet drops. Then there should be categories of malicious nodes such as pure malicious and collaborative malicious. Pure malicious nodes could misbehave individually, while collaborative malicious nodes misbehaving as a team in the network. Therefore, it was important to distinguish the type of malicious nodes.

## **Research Objectives**

* Identify malicious nodes in the network and keep continuing transactions only with trustworthy nodes.
* Distinguish pure malicious nodes and collaborative malicious nodes in the ad hoc network.
* Predict the secured routes in MANET using Q-Learning mechanism.

# METHODOLOGY

## **Spiral Model**

As the advanced categorizing of the malicious nodes, we had to go to the spiral model where we had the collaborative malicious node discovery process. In the spiral model mainly, there were three different phases.

Table 1.1: Backup Table

|  |  |  |  |
| --- | --- | --- | --- |
| Neighbor node | Trust Value | Time duration/ Backup time | Analyzed results |
|  |  |  |  |

### **Collaborative malicious node discovery process**

This was the phase where could do the advanced categorization for the malicious nodes and identified the collaborative malicious nodes by analyzing the dynamic behavior of the nodes. Only using one record it could not predict a collaborative malicious behavior, and it had to has more historical records or trust records. For this purpose, mainly, maintaining a backup table as in Table 1.1, where it stored the recent records of the trust table and each entry on the backup table was associated with a timeout. Initially, it had predetermined range for the trust with high trust value (HT) and low trust value (LT) and using the backup table records and current trust record it could compare the values against the time. For a given time period it could analyze the trust values, and after getting the analyzed report or plot, it could check for outliers within the given range HT – LT. If it contained any outliers or there were any sudden dynamic changes of the trust values it could suspect as a collaborative malicious node. Otherwise, it could be a pure malicious node without any dynamic changing behavior. The range could be changed according to the user specification.

Figure 2.1: Flow Chart for the Spiral Model

The procedure for the algorithm in Figure 2.1 was as follows.

**Procedure:** collaborative malicious node discovery algorithm (spiral model)

1: Get highest trust value and lowest trust value for the node for a given time range and marked them as value boundary for outliers

2: Then compare current trust value is in between the range or not.

3: If current trust value is in between the range, it's categorized as a pure malicious node.

4: Then it (the node who execute this) can delete that record from all of its tables and can broadcast message to aware others.

5: So that will terminate the pure malicious identified process.

6: If current trust value is not in between outliers it's categorized as collaborative malicious (CM) node.

7: Then it (the node who execute this) can edit its trust table blackList flag to true.

8: Identify all the neighbors of identified CM node and reduce their trust value since they have given the incorrect recommendations.

9: Broadcast to the other nodes

10: Go to the Identifying\_trust\_levels algorithm again.

### **Penalty phase**

Same as the trust level identification phase here also reduced the trust value of the particular recommending node with the help of a reduction factor. Reduction factor could be calculated based on the maturity level or the reputation of the node. Immediately after the trust reduction, old trust value in the trust table should be updated with the newly calculated value. According to the updated trust value, the particular neighbor nodes should be redirected to the trust level identification phase in order to re-categorize their trust levels. The pseudocode for this algorithm was as follows.

BEGIN

p\_M =passed-in malicious node

IF trust value is not an outlier

THEN

Delete from trust table

Send Broadcast to delete the node

ELSE

Mark node as a blacklist in trust table

FOR each node which recommended p\_M DO

Calculate reduce factor

Recalculate indirect trust

Update global trust

END FOR

Broadcast neighbors about p\_M node

GOTO Identifying\_trust\_levels

END IF

END

## **Deep Reinforcement Learning Model**

Reinforcement Learning (RL) model was trained to achieve a particular goal through the optimal path. It could assign a positive reward for corrective action and a negative reward for incorrect action. RL model could predict more accurate result without utilizing more historical data of the relevant scenario. As for the agent in the RL model, used a DQN (Deep Q Network). DQN was a network which consisted of state changing operations where an agent could monitor the actions and assign rewards for accurate actions and assign penalty points for inaccurate actions.

Figure 2.2: System Diagram

As in Figure 2.2, global trust value could be inputted to the RL component. Then it could generate a q-value based on defined rewards. This q-value could determine the most trustworthy path to forward packets. If the q-value was high then it could consider as the more trustworthy route and if q-value holding a law value then it could be an untrustworthy route. Here, assigned performance metrics as the state and predict actions according to it by the DQN agent.

## **Implementation**

### **Technologies**

In order to simulate the trust model on top of AODV routing protocol, used a popular network simulator called ns-3. For implementing the trust model, used c++ programming language and the entire RL model was implemented in python. Keras library was used for handling memory architectures or layers. Gnuplot was utilized for graphing backup table data with minimum and maximum global trust values. As for the development tools, Eclipse and Pycharm were used and the Github repository was the version control system. For maintaining coding styles, used the ns-3 standard code formatter.

# RESULTS AND DISCUSSION

## **Spiral Model**



Figure 3.1: Sample Network Topology

We took a topology of five network nodes including one **malicious node - 10.0.0.3** (black color node in Figure 3.1). Total simulation time was 100 seconds and calculating trust values in every 10,40 and 90 seconds. There the goal was to identify the category of the malicious node through the spiral model.



Figure 3.3: Backup Table Data for Node 2

Figure 3.2: Backup Table Data for Node 1





Figure 3.4: Backup Table Data for Node 3

Figure 3.5: Backup Table Data for Node 4



Figure 3.6: Backup Table Data for Node 5

Figure 3.2- Figure 3.6 represent the plotted backup table data for all the nodes in the above-defined network. Each diagram consists of minimum and maximum global trust value (outliers) of each neighbor node. Based on those outliers, it could identify that 10.0.0.3 is a pure malicious node.

## **Deep Reinforcement Learning Model**

In order to measure the trustworthiness of routes will analyze result of a flow wise approach which was generated by the flow monitor in ns-3 simulator. Following images represent q-values of each flow node by node separately.

Figure 3.8: Q-values for Flows in Node 2

Figure 3.7: Q-values for Flows in Node 1



Figure 4.0: Q-values for Flows in Node 4

Figure 3.9: Q-values for Flows in Node 3

We had to execute the simulation multiple times in order to train the DQN agent. After analyzing the result from Figure 3.7 to Figure 4.0, it was revealed that the second highest global trust holder is the most trustworthy node in this neural network.

# 4. CONCLUSION

This document describes proposed solutions, methodologies and techniques used in the spiral model and deep reinforcement learning model. It presents a novel solution to distinguish pure malicious nodes and collaborative malicious nodes using the spiral model. If the current global trust value is in between minimum and maximum global trust range, it could be a pure malicious node. If not, it is in the form of collaborative. After all the trust level classifications, RL component will determine the most trustworthy node according to defined DQN agent. Then we could conclude the trust model accuracy on top of AODV routing protocol.

# REFERENCES

[1] S. S, P. K, and T. A, “Authentication Using Trust to Detect Misbehaving Nodes in Mobile Ad hoc Networks Using Q-Learning,” *Int. J. Netw. Secur. Its Appl.*, vol. 8, no. 3, pp. 47–64, 2016.

[2]     K. J. R. Liu, “Information theoretic framework of trust modeling and evaluation for ad hoc networks,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 305–317, 2006.

[3]       R. Dalal, “Different Ways to Achieve Trust in MANET,” *Int. J. AdHoc Netw. Syst.*, vol. 2, no. 2, pp. 53–64, 2012.

[4] T. Farid and A. Prahladachar, “Secure Routing with AODV Protocol for Mobile Ad Hoc Networks.”

[5] S. Subramanian and B. Ramachandran, “QOS Assestion in MANET Routing based on Trusted AODV,” Int. J. Ad hoc, Sens. Ubiquitous Comput., vol. 3, no. 3, pp. 135–143, 2012.

[6] J. S. S.Uma, “Human Interaction Pattern Mining Using Enhanced Artificial Bee Colony Algorithm S.,” *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 3, no. 10, pp. 10131–10138, 2015.

[7] R. K. Bar, J. K. Mandal, and M. M. Singh, “QoS of MANet Through Trust based AODV Routing Protocol by Exclusion of Black Hole Attack,” *Procedia Technol.*, vol. 10, pp. 530–537, 2013.

[8]       C. Fountas, “Swarm Intelligence: The Ant Paradigm,” Springer, Berlin, Heidelberg, 2010, pp. 137–157.