ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.6

Flooding Attack Detection using Artificial Intelligence based IDS in VANET

Sheetal Panjeta[1], Er. Kanika Aggarwal[2]
PG Student[1], Assistant Professor[2]
Department of Computer Engineering
Maharishi Ved Vyas Engineering College, Yamunanagar, Haryana, India

Abstract:
Vehicular Ad hoc Networks (VANETs) are classes of ad hoc network and are created by applying the principles of mobile ad hoc networks (MANETs).VANETs provides communication among various vehicles and roadside units. These are decentralized and due to this, are susceptible to many security attacks. Flooding attack is one of the major security threats to VANET environment. These attacks mostly influence the five requirements- the availability, confidentiality, integrity, non-repudiation and authenticity of the system. So to tackle these attacks Intrusion Detection System (IDS) are used. This paper proposes a hybrid Intrusion Detection System which increases accuracy and other performance metrics using Artificial Neural Networks as classification engine and Genetic algorithm as optimization engine for feature subset selection. Various performance metrics are calculated and compared with other researchers work. The results attained show high accuracy and correctness and insignificant false alarm rate.

Keywords: RREQ Flooding, Intrusion Detection System, Artificial Neural Network, Genetic Algorithm, Security, VANETs

I. INTRODUCTION

Vehicular ad hoc networks (VANETs) are special category of MANETs but differ in its movement. . In MANETs the node can move randomly whereas in VANETs the node does not follow random movement. The nodes imitate like vehicle and move along the direction of roads. Due to increase in population, there has been rapid increase in number of vehicles. The increase in vehicles tends to increase the chance of road accidents. According to the survey, there has been 12 lakhs life are lost daily worldwide [1]. We need to have a technique by virtue of which the vehicles can be made smart enough so that they are able to handle the road safety at their own. This concept was the laid under VANETs to provide secure and reliable driving environment. VANETs allow mainly two type of interactions- V2I (vehicle to infrastructure) and V2V (vehicle to vehicle) as shown in Fig 1. Apart from this there is yet another interaction that takes care about essential information like fatigue detection of driver. This type of interaction is known as intra-vehicular interaction. VANETs are complied with IEEE 802.11p dedicated short range communication (DSRC) [2]. The vehicles have the On-Board Units (OBU) which consists of sensors. The communication has to be sent in form of cooperative awareness message (CAM) and has to pass through Road Side Units (RSU) [3].

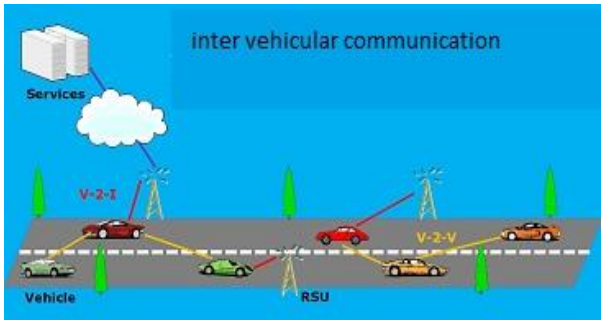


Figure.1. Communication in VANETS [5]

In VANETs the OBU is responsible for interacting with outside network which includes other vehicles and roadside unit infrastructure. VANETs have large number of applications. These include safety applications which let other vehicles know about the status of road and can protect from accident. There are also user based applications which entertain the user on drive where driver can download media files or access the weather condition etc[4]. VANETs are highly mobile and lack a fixed infrastructure. There is no guarantee of end to end connection [4].The auto configuration is one of its demerits. Along with large number of applications some involving life saving applications there are few challenges associated with VANETs such as high mobility, scalability and fault tolerance and the most crucial is the security. To handle these attacks there are two type of solutions- cryptography based solutions and Intrusion Detection Systems (IDS). In this paper we have used IDS based solutions as cryptographic solutions does not prove to be robust while finding new type of attacks and are also resource intensive. There are many types of attacks that can occur due to vulnerabilities in VANETs. We have focused on RREQ Flooding attack in which an intruder node tries to send multiple number of route request messages to a node which does not exist in network thereby consuming the channel that was supposed to be devoted to valid node for service. The rest of the paper is organized as follows. Section 2 discusses about the work done in the related field by various researchers. Section 3 gives an outline about Intrusion Detection System. Section 4 gives an overview of Artificial Intelligence techniques. Section 5 explains

the proposed work. Section 6 and 7 covers the result and conclusion part respectively.

II. RELATED WORK

The solutions to various security attacks are classified into cryptographic and IDS. Grover et al. [9] proposes solution to combat Sybil attack by the use of session keys and digital signature with sequence numbers. Zhou T et al. [6-7] proposed Public Key Infrastructure wherein each node has public-private keys. While sending the information the sender signs the message with its private key and add Certificate Authority (CA) certificate. The receiver verifies the signature. Another alternative proposed by Hao Y. et al. [8] was group signature but this is complex in nature as every time a vehicle enters the group its public key and vehicle session key has to be changed and transmitted to the group. Hoque et al. [10] proposed an intrusion detection system using genetic algorithm to detect different network intrusions. The famous KDD Cup99 dataset was used. The paper presented two phases- the first phase gave rise to new chromosomes by proving the network data. The second phase outputs the type of data whether attack or normal by taking previous phase as input. The detection rate gave remarkable results but can be improved with more hybrid techniques used in detection phase. Benaicha et al. [11] proposed IDS using GA. The features extracted from five different attack types and rules are formed for each of them. The paper builds 80 rules for each attack type which are then fed to GA model. After these 400 rules the evolution process takes place and fitness function is calculated and rule set are formed. Sen et al. [13] proposed back propagation Neural Network (BPNN). The BPNN is composed of multiple hidden layers. This paper performs 2 experiments one with 70-30 split and another with 80-20 split of dataset and uses different number of nodes in each hidden layer. The training is done on 1000 epochs and the number of hidden layers is fixed to 4. The number of features selected is 40 and then all of them are assigned a numeric value for normalization to take place and the confusion matrix is formed. Barati et al. [14] proposed a network based anomaly IDS to detect DDOS attack using GA and MLP of ANN. The ANN is composed of 3 layers. The data received from GA is passed to internal layer. The middle layer processes the data and the external layer gives the output. Saied et al. [15] proposed an intrusion detection system based on ANN to combat known and unknown DDOS attack. It used SNORT IDS to monitor the network. The IP identifier identifies the IP address and is then passed to ANN engine which compares it with existing pattern to detect attack. After the detection phase it is passed to defense phase whose role is to stop the attack and allow only legal packets to pass through. It also took the output from other snort ids as well to determine if it has an out dated algorithm in which case retraining is required. The last phase is knowledge share where each detector sends message to other ids. These messages were encrypted. The dataset uses 80% training and 20% testing set.

III. OVERVIEW OF INTRUSION DETECTION SYSTEM

As we saw, VANETs are susceptible to various security attacks, so there is need of some measures to tackle these attacks. We saw some of the solutions like group keys, encryption policies etc, but these solutions do not prove to be successful when a new attack is to be inspected. These solutions can only be used as entry level of protection. After these solutions there need to be another layer. This layer is of Intrusion Detection System (IDS). An IDS is hardware or software that tries to detect any abnormal behavior in the network.IDS operates in 3 phases as depicted in Fig 2. First one is event monitoring which includes collection of data for any abnormal behavior. Second one is analysis process which includes various techniques like statistics, pattern matching, machine learning etc. The last phase is response generation which detect the abnormal behavior and report to admin. In this paper we have used Artificial Intelligence approach to implement IDS.

First we have used Artificial Neural Networks (ANN) to detect the RREQ Flooding attack and later on uses Genetic algorithm (GA) to reduce the number of features from the dataset.

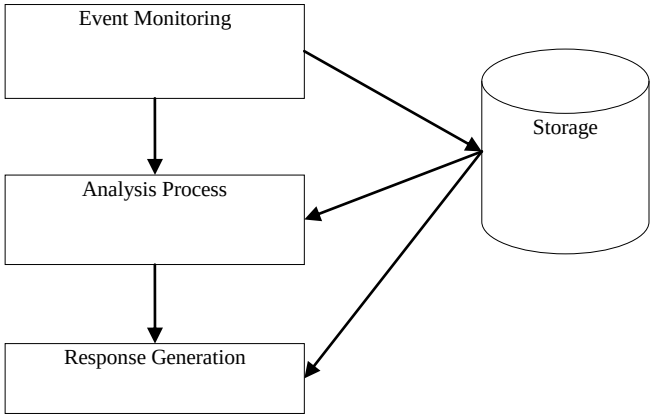


Figure.2. Architecture of IDS

IV. ARTIFICIAL INTELLIGENCE TECHNIQUE

A. *ARTIFICIAL NEURAL NETWORKS*

An artificial neuron network (ANN) is a branch of machine learning techniques and computational model based on the structure and functions of biological neural networks. ANN is a system of connected neurons where each neuron is connected to all other neurons. Neurons are the most important processing unit in ANN. The brain processes the information by interchanging the pulses among neurons. The neurons are connected to each and every input with weight associated with particular input and results in the output. The same idea is applied by ANN in computer science for classification or prediction based issues. ANN is flexible system by virtue of which the arrangement of network changes due to change in the inputs, weights associated with inputs or any other parameter. ANN can be described in two ways- single layer perceptrons (or single layer neural networks) and multi layer perceptron. The Multi Layer Perceptron consists of input layer, one or more hidden layer and finally the output layer. The number of hidden layer depends on the application. The hidden layer is also referred to as the processing layer. Fig 3 shows the architecture of multi layer perceptron with 1 input layer consisting of two neurons, 1 hidden layer with 3 neurons and an output layer with 1 neuron. The input layer consists of as many neurons as the number of features in data set. The processing takes place at hidden layer and finally the output is formed at the output layer. Each input is associated with a weight.

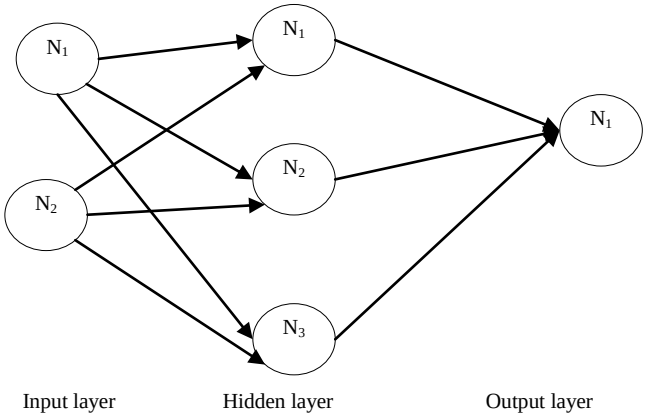


Figure.3. Architecture of ANN

B. *GENETIC ALGORITHM*

Genetic algorithm is an artificial intelligence heuristic approach which emulate the methodology of natural evolution. Evolution is the process by virtue of which the organisms improve consecutively over generations through the GA operators described later in this section. The Genetic Algorithm is used for optimization problems. It follows the principle of survival of the best which means the best feature individual will be selected over consecutive generations and hence improving and making the system more efficient.

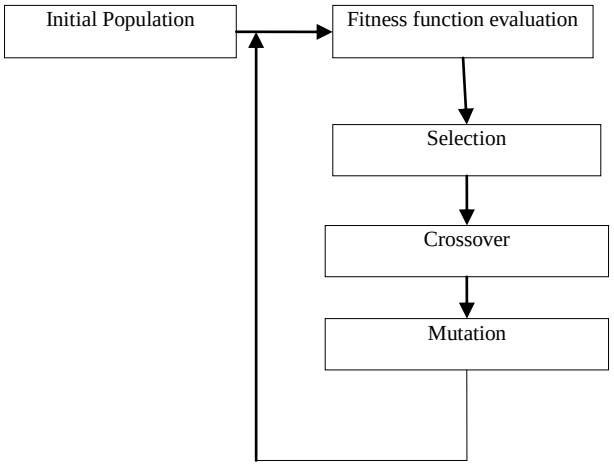


Figure.4. Genetic algorithm operations

The Genetic algorithm starts with an initial population. This initial population is randomly chosen from list of population. After initializing the initial population, the fitness function is evaluated. Based on the result of evaluation the genetic algorithm operators - selection, crossover and mutation are applied. The whole procedure is repeated until stopping criteria is matched [18]. Fig 4 shows the operations of genetic algorithm. In this paper we have used GA for feature subset selection

V. PROPOSED WORK

The proposed system is used to detect RREQ Flooding attack using ANN. It is optimized in terms of feature subset selection using GA. The network simulator ns-2.35 is used for launching the RREQ Flooding attack. This algorithm works well for multiple numbers of malicious nodes and gives extraordinary results on evaluating the performance metrics like accuracy and false positive rate. The simulation of VANET environment is done through SUMO. The tcl file which is generated by MOVE simulator is used as input to NS-2.35. The implementation generally involves three stages. The first stage is creation of dataset by launching the attack. The second step is Data Preprocessing and the last step is the classification and optimization engine. Fig 6 shows the proposed system architecture.

Creation of Data set: For creation of data set, a VANET environment was set up by integration of SUMO, MOVE and NS-2.35. The output of MOVE file is a tcl script used by NS-2.35. The output of trace files were collected as output for two different scenarios- normal AODV and AODV under RREQ Flooding attack. The purpose of this attack is to consume the network bandwidth and to exhaust the network resources all the time. Fig 5 shows the pseudocode for creation of data set.

Data Preprocessing: The real data consists of erroneous data and might also be not very useful as it is raw data. There is a need to convert this raw data into meaningful information. To achieve this objective, data preprocessing is required. The data preprocessing phase as described below in our proposed architecture involves three main steps- extraction of features from data set, data cleaning and data normalization.

```
Input: MOVE.jar
    Output: Two trace files normal.tr and attack.tr
1.  no_of_vehicles ←20
2.  simulation_time ← 200
3.  for each node n
4.          set node's cordinates
5.  end for
6.  for each edge e
7.          lanes ← 20
8.          speed ←30
9.          priority ←70
10.         initialize edge id
11. end for
12. for each flow f
13.         initialize flow id
14.         no_of _vehicle_per_flow
    ←no_of_vehicles/no_of_ flow
15. end for
16. CreateVehicle()
17. ConfigVehicle()
18. call Visualize()
19. add_Connection()
20. normal.tr ←NS2()
21. RREQ_Flooding()
22. Atta
23. ck.tr ←NS2()
24.
```

Extracting features from data set:

The normal.tr and malicious.tr files obtained while launching the

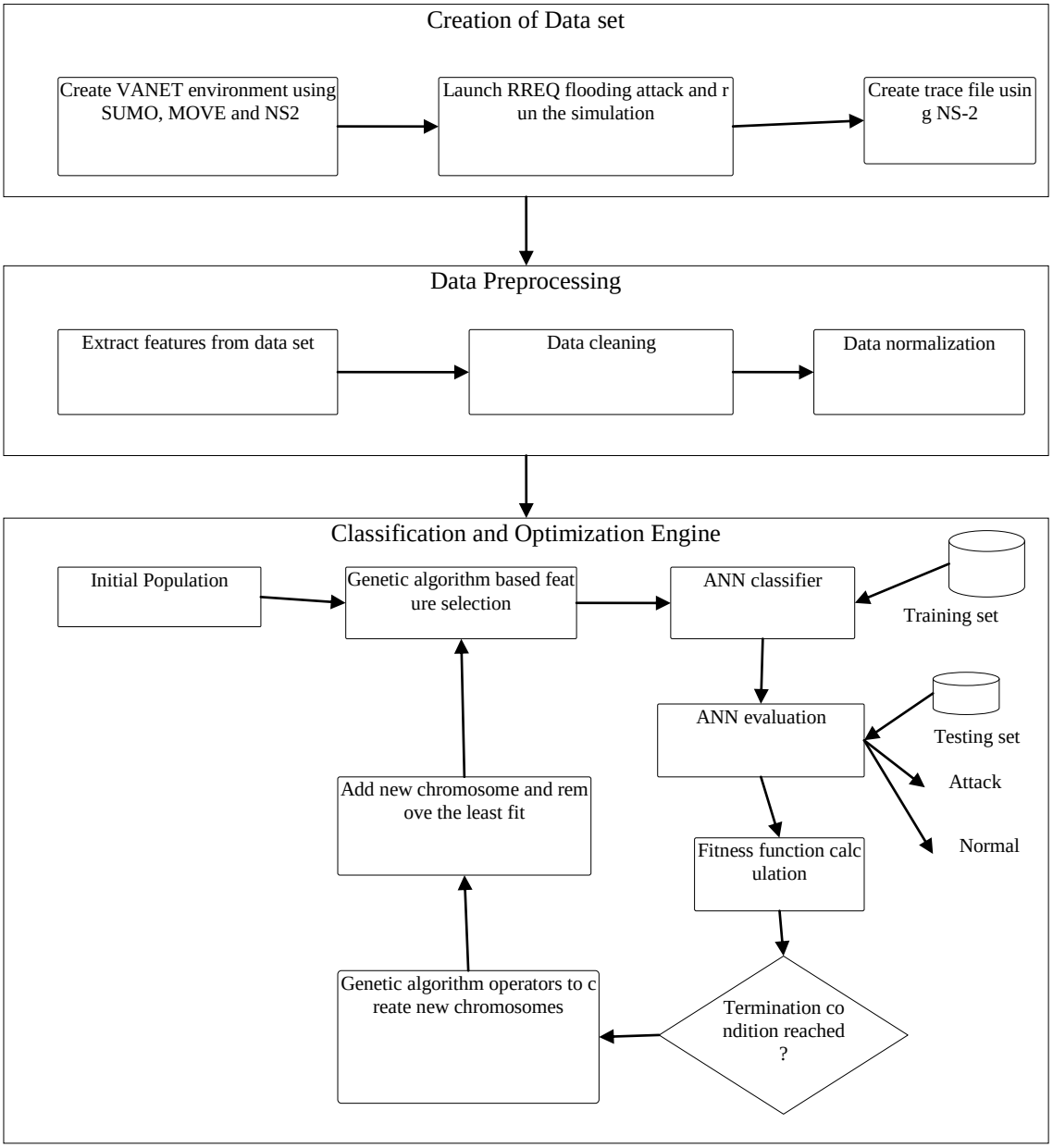


Figure.6. Proposed System Architecture

attack collectively forms a data set. These trace files contains many fields and are separated by space delimiter. The trace file is divided into three traces namely Basic Trace, IP trace and AODV trace.

Data Cleaning: It refers to correcting or removing of inconsistent records from the data set. In this phase we try to make the data set robust and unbiased. The redundant data or the data that assumes value zero all the time are removed. Handling of missing values is also done in this phase.

Data Normalization: This phase is further subdivided into two stages- Converting the values into numeric and the second stage is normalizing the data set so that the whole data set lies in one particular range.

Converting non numeric into numeric fields: In this phase the non integer values like TCP, AODV, MAC, RTR, AGT, etc are converted into integer. The columns which consist of hexadecimal values are also converted into integers. This phase is important to be performed for dataset to be normalized.

Converting data into standardized form: The dataset in the previous phase contains only integer values. But those integer values have a wide range. It would be selfish act to let that wide range of inputs. As when we apply the detection techniques the comparison between the fields would make no sense if they have huge deviation. The data with huge values will affect the data with lesser values; hence converting the data into standard form becomes indispensable.

Classification Engine: The ANN is used as classification engine which classifies the features into two main categories- the normal class and the attack class based on knowledge learnt in training phase. The inputs are passed to the hidden layer where the processing takes place with the help of transfer function and later the output of these are passed to next hidden layer (in case of multiple number of hidden layers) or to the output layer (in case of single hidden layer) which finally gives the output. Since the procedure is from first layer to last layer it is feed forward network. The back propagation technique is used to update the weights which are associated with each input in order to improve the training functions. This updation of weights is done from the outer layer to the inner layer, hence the name back propagation. Levenberg-Marquardt algorithm is used as back propagation and is given as

$$\mathbf{x}k+1=\mathbf{x}k-[\mathbf{J}T\mathbf{J}+\mu\mathbf{I}]-1\mathbf{J}T\mathbf{e}.$$

where x_k is current value, x_{k+1} is the updated value, I is the identity, e is the network error, J is jacobian matrix and μ is scalar. The whole process of training and testing which includes calculating the output and updation of weights is known as feed forward back propagation network. Fig 7 shows the workflow of classification engine.

Optimization Engine: The GA is used as optimization engine to reduce the number of features. The initial population is randomly chosen and feature subset selection is applied on it. It is passed to the ANN classifier. The classification accuracy is considered as fitness function. The fitness value of each feature set chromosome is calculated. The features which do not contribute in enhancing the accuracy of the system are dropped. In this paper we have used wrapper method for subset selection. The best set of features are retained and the least one is removed from the list hence modifies the feature set. Fig 7 shows the pseudocode for the same.

```
Input: Intial population
Output: Feature sub selection
1.  do { For each chromosome c
2.          Calculate fitness function
3.          Create vector using mutation
4.          Create new individual using
    crossover
5.  end for
6.  if fitness value is greater than previous
    individual
7.          add the individual to the list
8.  else
9.          remove the least fit individual
10. }while(stopping criteria is not met)
```

Figure.7. Pseudo code for feature subset selection

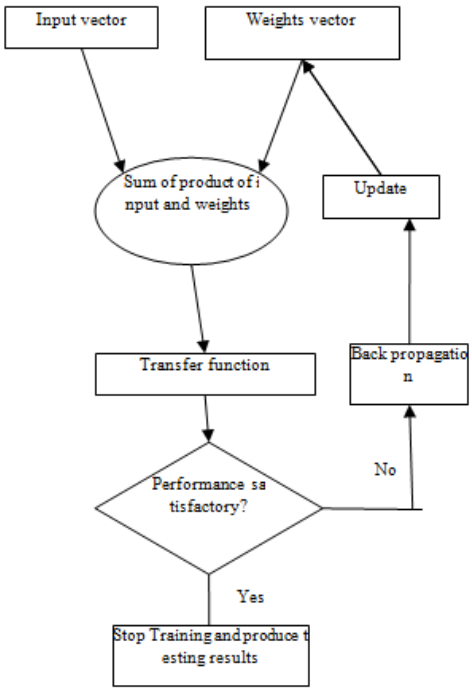


Figure.8. Workflow of ANN along with back propagation

VI. RESULTS AND DISCUSSION

The performance metrics have been evaluated in two scenarios- misuse detection and anomaly detection. In both the scenarios 10 fold cross validation is used. The whole dataset is split into ten equal subsets with 10% records set for testing in each dataset. The average of all these results has been taken into consideration to avoid any biased results. The genetic algorithm has been implemented for feature sub-selection. The final number of features left in the system is 18. The performance metrics are calculated using the four parameters- True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The accuracy of system is evaluated as

Accuracy= (1)
Precision= (2)
Specificity= (3)
Sensitivity= (4)
False Positive Rate= (5)
False Negative Rate= (6)

Table I, Table II and Table III shows the parameter settings for simulation environment, ANN and GA respectively.

Table.1. Simulation Environment Parameters

Parameter	Value
No of nodes	20
No of malicious nodes	2
Channel Type	Wireless
Routing Protocol	AODV
MAC_TYPE	802.11
Packet Size	1000
Interface Queue Type	Queue/Drop Tail/Priority
Simulation Time	200

Table. 2. ANN Parameters

Parameter	Value
Network Type	Feed forward backdrop
Training Function	Levenberg-Maquardt
Performance Function	Mean Square Error
No of layers	2
No of neurons per layer	10
Transfer Function	Transig
Maximum Epochs	1000
Validation check	6
Data Division	Random

Table. 3. GA Parameters

Parameter	Value
Genetic Operations	Scattered crossover, single point mutation
Selection Method	Stochastic
Crossover Rate	0.8
Mutation Rate	0.02
Population Size	200
Max Generations	100
Stall Generations	50

The various performance metrics have been evaluated and the outcome is shown in Table IV and Table V. Table IV shows the results for misuse detection and Table V shows the result for anomaly detection. A comparative analysis of our approach with existing algorithms has been shown in Table VI. Our approach shows remarkable results especially in terms of accuracy and false positive. A graphical comparison of accuracy of our approach with other existing approaches has been shown in Fig 10.

Table.4. Results (misuse)

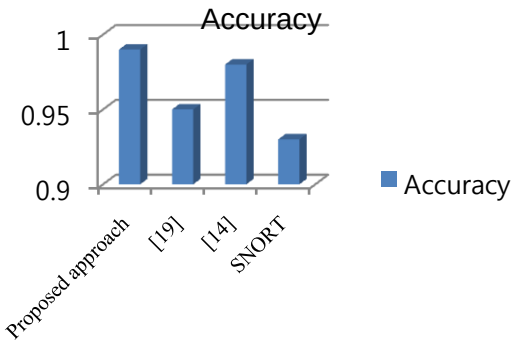
Precision	1
Specificity	1
Sensitivity	0.99
Accuracy	0.99
F_measure	1
False Positive Rate	0

Table.5. Results (Anomaly detection)

Precision	0.97
Specificity	0.97
Sensitivity	0.99
Accuracy	0.95
F_measure	0.98
False Positive Rate	0.03

Table.6. A comparison of different performance metrics of our approach with other existing approaches

Algorithms	Precision	Specificity	Sensitivity	Accuracy	F_measure	False Positive
Proposed approach	1	1	0.99	0.99	1	0.01
ANN based driverless car [12]	NA	.87	.98	NA	NA	0.12
ANN and Decision Tree based IDS [19]	NA	NA	NA	0.95	NA	0.10
ANN based IDS [14]	1	1	0.96	0.98	NA	NA
SNORT	0.96	0.97	0.9	0.93	NA	NA
GA and MLP based IDS [15]	1	NA	0.99	NA	0.99	0.03



VII. CONCLUSION

As security is essential part of wireless networks especially vehicular networks, there was a need to handle the threats which could occur in vehicular networks. One of the most serious security threat is RREQ flooding in which the legal user is denied to get the service due to unavailability of resources. In our paper we have launched the flooding attack in ns-2 and evaluated the packet delivery and throughput of the network. There was sharp fall in packet delivery and throughput. We proposed our algorithm based on ANN to detect the attack and further applied GA for feature sub-selection to obtain better results under two different situation misuse and anomaly. Our proposed algorithm can detect multiple malicious nodes with higher accuracy as compared with existing approaches. The accuracy of our system came to 99%. Moreover the number of features was reduced to 18. There is no need for any hardware, hence simple and cost effective. This paper showed remarkable results but the future scope lies in detecting the attacks with encrypted malicious entries. Also the data set used is specifically for flooding attack. We would like to extend our proposed algorithm to make it more generic by adding more records of other attacks as well.

VIII. REFERENCES

[1]. R. S. Raw, M. Kumar, and N. Singh, "Security challenges, issues and their solutions for VANET". *International Journal of Network Security & Its Applications*, vol 5, No. 5, pp. 95, 2013

[2]. B. Patel and K. Shah, "A Survey on Vehicular Ad hoc Networks," *IOSR Journal of ComputerEngineering (IOSR-JCE)*, vol. 15, no. 4, pp. 34-42, 2013

[3]. K . M Ali Alheeti, A. Gruebler and K. D McDonald-Maier, "An Intrusion Detection System against malicious attacks on the communication network of driverless cars," in *2015 12th Annual IEEE on Consumer Communication and Networking Conference (CCNC)*, Las Vegas, 2015.

[4]. M . H Kabir, " Research issues on Vehicular Ad hoc Network," *International Journal of Engineering Trends and Technology*, vol. 6, no. 4, pp. 174-179, 2013

[5]. S. Dhankhar and S. Agrawal," Vanets: A survey on routing protocols and issues."*International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*, vol3, No.6, pp.13427–13435, 2014

[6]. T. Zhou,et.al, "P2dapsybil attacks detection in vehicular ad hoc networks". *Selected Areas in Communications, IEEE Journal*, vol 29,No.3, pp:582–594, 2011

[7]. T. Zhou et.al, "Privacy-preserving detection of sybil attacks in vehicular ad hoc networks". In *Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, Philadelphia , pp: 1–8. IEEE, 2007

[8]. Y. Hao, et.al, "A distributed key management framework with cooperative message authentication in VANETS". *Selected Areas in Communications, IEEE Journal*, vol 29,No.3, pp::616–629, 2011

[9]. J. Grover, M. S. Gaur, and V. Laxmi, "A novel defense mechanism against sybil attacks in VANET". In *Proceedings of the 3rd international conference on Security of information and networks*, pp: 249–255. ACM, 2010.

[10] .M.S. Hoque, M. Mukit, Md Bikas. "An implementation of intrusion detection system using genetic algorithm". *International Journal of Network Security & Its Applications (IJNSA) vol 4, No.2, pp:109-120, 2012*

[11]. S. E. Benaicha et.al, "Intrusion detection system using genetic algorithm". In *Science and Information Conference (SAI), 2014*, London pp: 564–568. IEEE, 2014

[12]. K. M. A. Alheeti, A. Gruebler, and K. D. McDonald-Maier, "An intrusion detection system against malicious attacks on the communication network of driverless cars". In *Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE*, Las Vegas, pp: 916–921. IEEE, 2015

[13]. N.Sen, R.Sen, and M. Chattopadhya, "An effective back propagation neural network architecture for the development of an efficient anomaly based intrusion detection system." In *Computational Intelligence and Communication Networks (CICN), 2014 International Conference Bhopal*, pp. 1052–1056. IEEE, 2014.

[14]. M. Barati, et.al, "Distributed denial of service detection using hybrid machine learning technique". In *Biometrics and Security Technologies (ISBAST), 2014 International Symposium Kuala Lumpur*, pp 268–273. IEEE, 2014.

[15]. A. Saied, R. E. Overill, and T. Radzik,. "Detection of known and unknown ddos attacks using artificial neural networks". *Neurocomputing*, vol 172, pp:385–393, 2016.

[16] R.Mitchell & R. Chen, "A survey of intrusion detection in wireless network applications," *Computer Communications*, vol. 42, pp. 1-23, 2014

[17]. E. Balkanli, J. Alves, and A. N. Zincir-Heywood. "Supervised learning to detect ddos attacks. In *Computational Intelligence in Cyber Security (CICS), 2014 IEEE Symposium*, Orlando, pp 1–8. IEEE, 2014

[18]. D. Pal and A. Parashar, "Improved genetic algorithm for intrusion detection system". In *Computational Intelligence and Communication Networks (CICN), 2014 International Conference Bhopal*, pp: 835–839. IEEE, 2014

[19]. S. Selim, M. Hashem, and T. M. Nazmy,. "Hybrid multi-level intrusion detection system". *International Journal of Computer Science and Information Security*, vol 9, No.5, pp::23, 2011