Improvement in Intruder Detection in MANET using Selected Features from NSL-KDD Dataset

[1]Shalu Verma, [2]Mrs. Nisha Charaya

[1]M.Tech Scholar, Computer Science and Engineering, Om Institute of Technology and Management, Juglan (Hisar)

[2]Assistant Professor, Computer Science and Engineering, Om Institute of Technology and Management, Juglan (Hisar)

Abstract: MANET (Mobile ad hoc network) is a collection of mobile nodes without a centralise controlling node. The intruder in the network can harm the whole architecture due to absence of controlling watchdog. The machine learning algorithms comes handy for this. Every MANET node can be deployed with ML algorithm to detect the intruders. For this previous behaviour of malicious nodes has to be studied. We used NSL-KDD public data set for MANET intruder detection in our work. In this work we optimally selected the features to feed into machine learning model of SVM by gravitational search algorithm.

I. INTRODUCTION

MANET is a very dynamic and continuously changing ad-hoc network, so to have a centralise monitoring on it is not possible. VANET is like MANET in which vehicles keeps on communicating with nearby vehicle and road side unit. It is highly dynamic in nature. To detect the intruder in it is very challenging task. An intrusion detection system (IDS) monitors network traffic and alerts the system or network administrator. IDS may also respond to anomalous traffic by blocking the user or source IP address from accessing the network. [1]

Some environments (such as the military tactical operations) have very stringent requirements on security, which make the deployment of security-related technologies necessary. Intrusion prevention measures, such as encryption and authentication, can be used in MANETs to reduce intrusions, but cannot eliminate them. For example, a physically captured node that carries the private keys may allow the defeat of the authentication safeguards. The history of security research has demonstrated that no matter how many intrusion prevention measures are used, there are always some weak points in the system [1][4]. In a network with high security requirements, it is necessary to deploy intrusion detection techniques. MANET IDSs, serving as the second wall of defence to protect MANETs, should operate together with prevention mechanisms (authentication, encryption etc.) to guarantee an environment with highsecure requirements. They should complement and integrate with other MANET security measures to provide a high-survivability network. However, most of today's Intrusion Detection Systems (IDSs) focus on wired networks. The dramatic differences between MANETs and wired networks make it inapplicable to apply traditional wired ID technologies directly to MANETs. MANET does not have a fixed infrastructure. While most of today's wired IDSs, which rely on real-time traffic parse, filter, format and analysis, usually monitor the traffic at switches, routers, and gateways. The lack of such traffic concentration point makes traditional wired IDSs inapplicable on MANET platforms. Each node can only use the partial and localized communication activities as the available audit traces. There are also some characteristics in MANET such as disconnected operations, which seldom exist in wired networks. What's more, each mobile node has limited resources (such as limited wireless bandwidth, computation ability and energy supply, etc.), which means MANET IDSs should have the property to be lightweight. All of these imply the inapplicability of wired IDSs on the MANET platform. Furthermore, in MANETs, it is very difficult for IDSs to tell the validity of some operations. For example, the reason that one node sends out falsified routing information could be because this node is compromised, or because the link is broken due to the physical movement of the node. All these suggest that an IDS of a different architecture needs to be developed to be applicable on the MANET platform.

Intrusion Detection in MANET's or adhoc networks is the task which can be related to machine learning field. The data set is available with NSL-KDD data for intrusion detection. On the basis of which forthcoming intruder whether that can be selfish node in the network, any malicious node or any Sybil node, can be detected as anomaly node. The dataset consists of previous history of intruders which are field names and their numerical values. This dataset is very large so dimensionality reduction has to be performed to select the best suitable features which gives highest accuracy and also consumes less time in training the model.

II. PROPOSED ALGORITHM

Our work is mainly targeted to feature reduction to get maximum accuracy and reduce the time overhead over classical machine learning algorithms with same complete features set. Intrusion dataset is taken from standard NSL KDD dataset from website of university of New Brunswick (UNB). This dataset is already explained in previous chapters. This dataset is very large having 125973 training data set and 22543 testing data set with total 41 features out of which 3 features are symbolic and output is subcategories of major types of attacks. Out of these 41 features, all of them don't contribute to

the accuracy of algorithm. Previously genetic algorithm (GA)-a metaheuristic algorithm was used to select the optimal features for enhanced accuracy but in our work we have replaced GA with Gravitational Search Algorithm (GSA) due to the property of GA to stuck into local minima which may result in skipping of some minima points over which accuracy can be highest. GSA is a global optimisation technique which checks every minima point to get the highest accuracy point in the search space. The search space for GSA is between 0 and 1 and GSA agents must lie either at 0 or 1 which means either that particular feature is considered or not.

Gravitational search algorithm is based on movement of planetary bodies, called as agents, which try to attract each other with a gravitational force. The orbit of these agents is the searching space for our problem which is in between 0 and 1 and all those agents are sitting at boundaries. Each agent's position is described by 41 variables or co-ordinates of agent's position are 41 in numbers. Since this is a binary gravitational algorithm for our case, there is a matrix with dimension equal to number of tuning variables (which are 41 features of NSL KDD dataset) with elements 0 and 1. There can be any number of agents in a searching space. For each agent an objective function is called which chose only those features out of 41 for which index value is 1. Multiclass SVM classifier is used to check the accuracy after dividing the data into set of 80% for training and 20% for testing. This accuracy for each agent is saved into a matrix and maximum of them is chosen. The initial position of agents are chosen randomly and further updated by adding the agents' velocity of movement into the old position. The velocity is dependent upon the fitness value (accuracy) calculated for each agent. Mathematically it can be formulated as:

$$x_i^d(t+1) = v_i^d(t+1) + x_i^d(t) \quad \dots (2.1)$$

$$v_i^d(t+1) = rand_i x v_i^d(t) + a_i^d(t) \quad \dots (2.2)$$

where $x_i^d(t+1)$ is the new agent's position for the next iteration and $x_i^d(t)$ is the present position. $v_i^d(t+1)$ is the new velocity of movement and $a_i^d(t)$ is the present acceleration. This can be further calculated as:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \dots\dots (2.3)$$

$F_i^d(t)$is the total force acting on ith agent calculated as:

$$F_i^d(t) = \sum_{j \in kbest} rand_j F_{ij}^d(t) \quad \dots\dots (2.4)$$

$F_{ij}^d(t)$Can be computed as:

$$F_{ij}^d(t) = G(t).\left(M_{pi}(t) \times \frac{M_{ai}(t)}{R_{ij}(t)} + \varepsilon\right).\left(x_j^d(t) - x_i^d(t)\right) \quad \dots\dots (2.5)$$

$M_{ai}(t)$ is the mass of an agent which is normalised accuracy value for each agent. It is formulated as:

$$m_i(t) = \frac{fit(t) - worst(t)}{best(t) - worst(t)} \quad (2.6)$$

where $fit(t)$ is the fitness value of each agent, $worst(t)$ is the minimum accuracy value among all present agents and $best(t)$ is the maximum accuracy value.

Following equations 2.1 to 2.6, every agent will get a new set of 0's and 1's. For this new set accuracy is again calculated following the procedure previously discussed. This is an iterative process and keep on repeating till all iterations. The agent's position for which accuracy comes out be maximum will be our reduced features set (neglecting the features with position index 0). A flow chart of complete process is shown in appendix A.1.

III. RESULTS

We have used MATLAB as a tool to simulate our proposal. The NSL-KDD dataset has four categories of intruder attacks which are: Denial of service attack (DoS), User to root attack (U2R), Remote to Local Attack (R2L), Probing attack. We have tested the algorithm on each attack separately converting the multilevel classification problem into binary classification.

case1: DOS attack

The data samples under this attack have six subcategories of attacks which are back, land, Neptune, smurf, pod and teardrop. So there are six classes again and classification problem has again converted into the multiclass. We optimized the number of features selected using GSA and previously used GA. A convergence curve for GSA is plotted which is accuracy vs iterations.

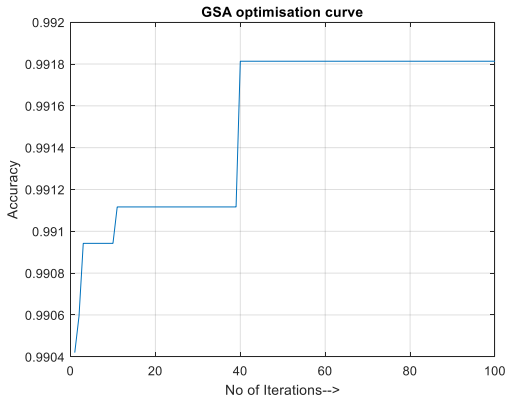


Figure 3.1: GSA convergence curve for DOS attack

The maximum accuracy is 99.18% upto which this algorithm converges for selected features.

Case II. Probe Attack

Probe attack is further subcategorized as 4 types namely ipsweep, nmap, portsweep and satan. The convergence curve for GSA in this attack is shown in figure 3.2. The saturation point in it is 0.9385 which is lesser as compared to DOS attack accuracy. An accuracy comparison between GA selected features and GSA selected features is shown in figure 3.3 for all subclasses in this attack. GSA selected features gave more accuracy compared to GA selected which is 3.3% more than GA.

Case III. User to Root (U2R) Attack

U2R attack is further subcategorized as 4 types namely bufferoverflow, loadmodule, perl and roortkit. The accuracy comparison is shown in figure 3.4. The accuracy in the attack detection is approx 54% by GSA method whereas it is 48% by GA optimized features.

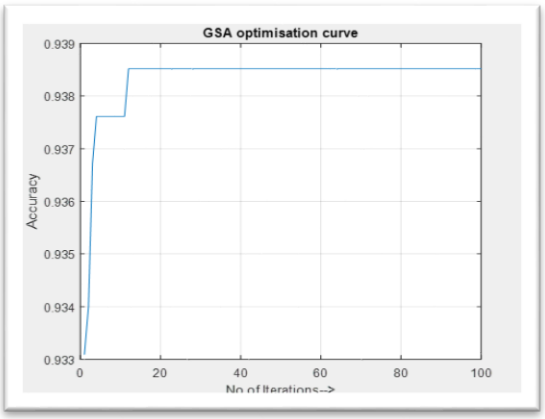


Figure 3.2: GSA Convergence curve for PROBE attack

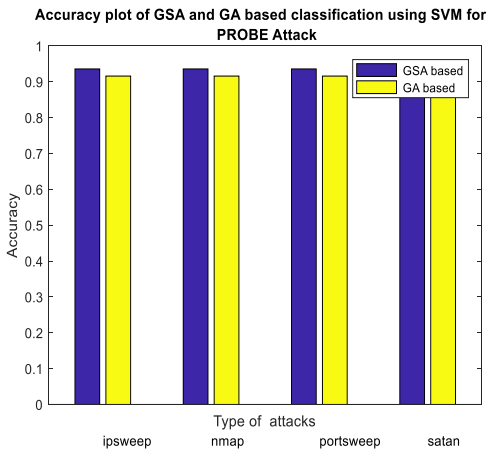


Figure 3.3: Accuracy comparison for all four sub attacks in PROBE attack using GA and GSA

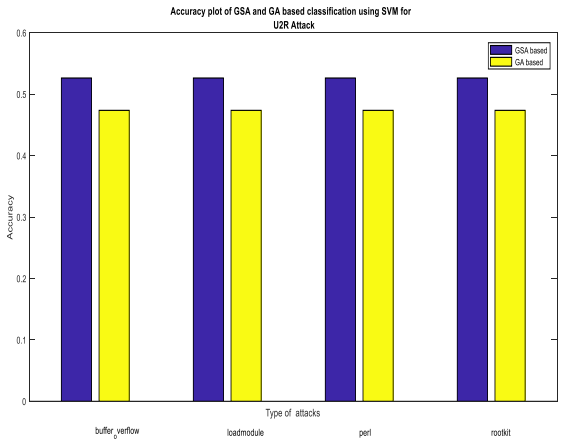


Figure 3.4: Accuracy comparison for all four sub attacks in U2R attack using GA and GSA

Case IV: Remote to User (R2L)Attack:

R2L attack is further subcategorized as 8 types namely ftpwrite, guesspasword, imap, multihop, phf, spy, warezclient, warezmaster.

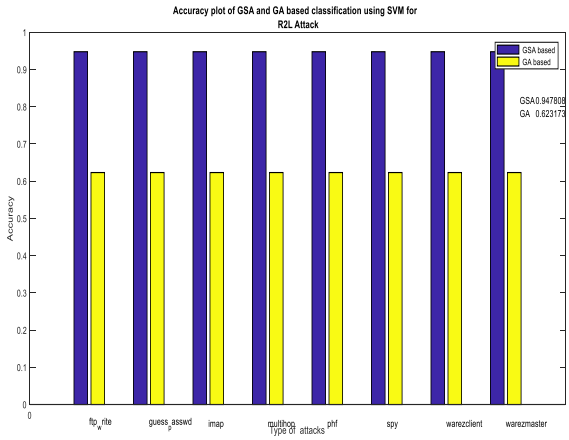


Figure 3.5: Accuracy comparison for all four sub attacks in R2L attack using GA and GSA

The accuracy in this attack for GSA selected features is 97% which is more than 53% from GA optimized features.

In every attack type GSA selected those features which gave more accuracy than GA selected features. A complete comparative table is shown in appendix A.2.

IV. CONCLUSION

MANET due to its consistently changing architecture, is very vulnerable to attacks. To prevent these attacks, detection of its nature is very important. Machine learning is the field which present the solution to this problem. For ML to use, we need to

have a database of previous attack history to know their behaviour. Our work used public available NSL-KDD dataset which is more managed and filtered version of KDD-cup dataset. All features in the data doesn't take part in detection and some may even pose a threat to go attack undetected. Our work is focused to remove those features and select only those which improve the detection accuracy. Gravitational Search Algorithm (GSA) is used which optimally chose only those features which actually takes part in attack detection. We tested the results for all four categories of intruders in the dataset and compared with genetic algorithm (GA) and we managed to get a maximum accuracy improvement of 54% in case of R2L attack. In all four categories, optimally selected features by GSA due to its global nature, gets higher accuracy than GA only.

V. REFERENCES

[1] Z. Ullah, M. S. Khan, I. Ahmed, N. Javaid and M. I. Khan, "Fuzzy-Based Trust Model for Detection of Selfish Nodes in MANETs," *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, Crans-Montana, 2016, pp. 965-972.

[2] M. A. Abdelshafy and P. J. B. King, "Dynamic source routing under attacks," *2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM)*, Munich, 2015, pp. 174-180.

[3] C. Alocious, H. Xiao and B. Christianson, "Analysis of DoS attacks at MAC Layer in mobile adhoc networks," *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*, Dubrovnik, 2015, pp. 811-816.

[4] A. Quyoom, R. Ali, D. N. Gouttam and H. Sharma, "A novel mechanism of detection of denial of service attack (DoS) in VANET using Malicious and Irrelevant Packet Detection Algorithm (MIPDA)," *International Conference on Computing, Communication & Automation*, Noida, 2015, pp. 414-419.

[5] A. M. Shabut, K. P. Dahal, S. K. Bista and I. U. Awan, "Recommendation Based Trust Model with an Effective Defence Scheme for MANETs," in *IEEE Transactions on Mobile Computing*, vol. 14, no. 10, pp. 2101-2115, Oct. 1 2015.

[6] A. Menaka Pushpa and K. Kathiravan, "Resilient PUMA (Protocol for Unified Multicasting through Announcement) against internal attacks in Mobile Ad hoc Networks," *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Mysore, 2013, pp. 1906-1912.

[7] M. A. Abdelshafy and P. J. B. King, "Analysis of security attacks on AODV routing," *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*, London, 2013, pp. 290-295.

[8] A. M. Kurkure and B. Chaudhari, "Analysing credit based ARAN to detect selfish nodes in MANET," *2014 International Conference on Advances in Engineering & Technology Research (ICAETR - 2014)*, Unnao, 2014, pp. 1-5.

[9] S. Biswas, P. Dey and S. Neogy, "Trusted checkpointing based on ant colony optimization in MANET," *2012 Third International Conference on Emerging Applications of Information Technology*, Kolkata, 2012, pp. 433-438.

[10] D. Das, K. Majumder and A. Dasgupta, "A game-theory based secure routing mechanism in mobile ad hoc network," *2016 International Conference on Computing, Communication and Automation (ICCCA)*, Noida, 2016, pp. 437-442.

[11] T. Poongothai and K. Duraiswamy, "Intrusion detection in mobile AdHoc networks using machine learning approach," *International Conference on Information Communication and Embedded Systems (ICICES2014)*, Chennai, 2014, pp. 1-5.

[12] D. A. Varma and M. Narayanan, "Identifying malicious nodes in Mobile Ad-Hoc Networks using polynomial reduction algorithm," *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, Chennai, 2016, pp. 1179-1184.

[13] Bandana Mahapatraa and Prof.(Dr) Srikanta Patnaik, "Self Adaptive Intrusion Detection Technique Using Data Mining concept in an Ad-Hoc Network," 2nd International Conference on Intelligent Computing, Communication & Convergence(ICCC-2016)

[14] Manjula C. Belavagi and BalachandraMuniyal, "Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection," 12th International Multi-Conference on Information Processing-2016 (IMCIP-2016).

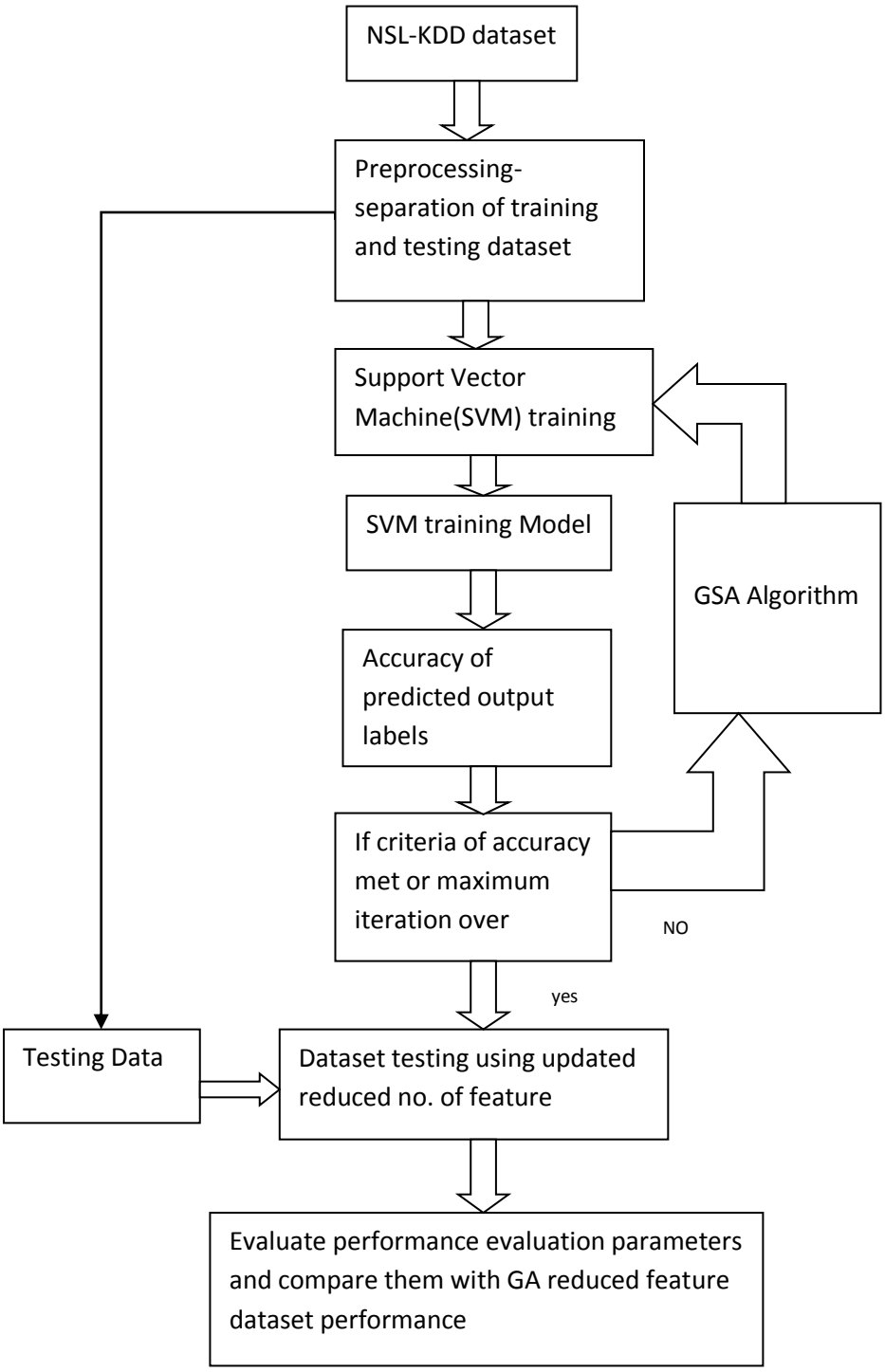
[15] PreetiAggarwala and Sudhir Kumar Sharmab, "Analysis of KDD Dataset Attributes - Class wise For Intrusion Detection," 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015).

[16] Ciza Thomas, Vishwas Sharma and N. Balakrishnan, "Usefulness of DARPA Dataset for Intrusion Detection System Evaluation

[17] P.Natesan and P.Balasubramanie, "Multi Stage Filter Using Enhanced Adaboost for Network Intrusion Detection," International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.3, May 2012

[18] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, ON, 2009, pp. 1-6.

Appendix A.1 Flow chart of the proposed Algorithm



Appendix A.2: Comparative results

Type of attack	Method used	Mean Value of parameters					
		Accuracy	Sensitivity	Specificity	Precision	Recall	F_measure
DoS	GA	0.981806452	0.5626428	0.981430543	0.468391874	0.5626428	0.586844502
	GSA	0.991483871	0.641325037	0.962736294	0.556824276	0.641325037	0.64520049
Probe	GA	0.648994516	0.517583399	0.63811076	0.31355712	0.517583399	0.465216237
	GSA	0.810133194	0.693804705	0.8036027	0.491220126	0.693804705	0.693735438
R2L	GA	0.606471816	0.239970672	0.605106568	0.158817724	0.239970672	0.235576288
	GSA	0.636743215	0.332945687	0.634314249	0.167813142	0.332945687	0.313969807
U2R	GA	0.684210526	0.25	0.574857026	0.171052632	0.25	0
	GSA	0.696315768	0.28	0.607485703	0.191052632	0.29	0