Anomaly Based Intrusion Detection Using Filter Based Feature Selection on KDD-CUP 99

Dr. R. Lalu naik[1], K. Samkeerthi [2]

[1]*Prof, Dept of CSE, Tirumala Engineering College, Narasaraopet, Guntur, A.P., India*

[2]*PG Scholar, Dept of CSE, Tirumala Engineering College, Narasaraopet, Guntur, A.P., India*

ABSTRACT - Data mining applications have become important in defending public and private computing systems from interruptions and security threats due to their use in recent years. The contemporary computing world is beset with numerous internal and external challenges. Interruptions may occur either in person or over a distance, and the typical techniques used to recognize them aren't enough to satisfy the need to predict and recognize interruptions. To prevent protection and reliability problems, detection of various attacks depends on the position of the interruption. Data mining is one of the best methods to identify and prevent infiltration in the modern world of security. This approach uses the most reliable and cost-effective mechanisms to give better results in data mining. Network security equipment and software, the intrusion detection system (IDS), is needed to protect the computing systems and monitor and identify network traffic data packets. An accessible, open-source network security technology called Snort IDS is available. However, the Snort program can only identify previously detected assaults. In the plan, the tenets of Data Mining are being used to help increase the IDS system's execution and to combat Differing processes such as Dynamic Data Preparation (DDP), Hybrid Rule-based Pre-preparing, and Simple K Nearest Neighbors Classification (SKNN) is used to describe some of the concerns including information preparation, pre-processing of the information, information order, and intrusion recognition.

Keywords: Intrusion detection, Machine learning, Cyber security

I. INTRODUCTION

The modern internet-based computer world has seen the internet and internet-based application usage increase fast, which an equal rise in cyber threats has mirrored. The administration is dealing with a tough challenge with the processing of new types of infiltration, and it has become a global concern.

An intrusion detection system (IDS) monitors systems and networks to catch any unauthorized activity or abuse. In 1980, James P. Anderson proposed various methods to improve security, auditing, and surveillance at client sites [2, 6]. Peter Neumann and Dorothy Denning built the first real-time IDS, which they termed the Intrusion Detection Expert System, between 1984 and 1986. (IDES). Initially, the now-named Next-Generation IDS (NIDES) was trained to detect known harmful behavior with a rule-based approach. This was enhanced and supported by the University of California and the U.S. government for programs like Haystack (U.S. Air Force). Work was done by comparing audit with known patterns, which led to the development of the Host-based Pattern matching system, which was then incorporated into the Distributed atmosphere (i.e., Distributed IDS). Todd Heberlein, a UC Davis professor, created NIDS (Networks Based Intrusion Detection Systems) in 1990. It was later adopted by DIDS (Detection and Discovery System). It was then deployed to develop the NSM (Network Security Monitoring) and similar commercial IDS like CMDS (Computer Misuse Detection System). ASM, a system for automated security measurement, was introduced to the market in 1994.

None of them has risen to the task despite the incredible advances in information production, detection procedures, and intrusion detection frameworks. The annual frequency of hacking and interference scenes is going up as progress is made. The danger of being breached is as much from outside interlopers as from those within the company. The firewall will likely break the framework, allowing the framework to be penetrated and unusable in separating positive or negative movement. In such cases, a firewall that is static and can't resist intrusion attempts is required. Intrusion Detection Systems can observe the activity of an opposing party on their systems. However, intrusion detection systems can detect breaches in security. Figure 1.1 illustrates the traditional depiction of IDS.

The fundamentals of the current digital age, as it relates to IDS, are system security design. To know the relevance of IDS, they must be aware of the interruption. When it comes to integrity, privacy, and openness, the interruption can be sorted. An activity or event triggers a breach of the framework's secrecy. A party or activity is considered a breach of integrity if it permits any motion in the conditions of obtaining the assets, such as money in a machine. A similar event may lead to an infringement of openness. For example, real customers may be prevented from accessing or taking advantage of the software's tools and resources if they access them via a computer. IDS can monitor the framework and the web to identify activity and then analyze its impact. IDS will work as a product or instrument for assault review or investigation, thus providing the chance to break down all occurrences. Research projects have envisioned a variety of disruption sites in light of the fast assault speed. The former framework and the existing structure are a handful of significant components, while the remainder moves away from the original design. See Figure 1.1 for the IDS structure that is conventional. Figure 1.1 shows some of the prominent groups of people who were raped and abused. It is intended to find episodes and strategies that information is liable to change due to changes in its organization. IDS' primary element is the Feature Extraction unit. A warning is issued to detect a few meddlesome actions.

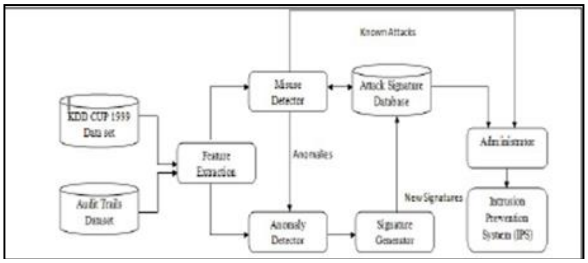

Figure 1: Representation of Generic intrusion detection system

II. RELATED WORK

J. Anderson suggested an interruption detection system in 1980 [1]. W. R. Cheswick[2] required three main types of existing firewalls for the portals linked to application entrance, packet data filtering, and network isolation, and this could lead to more. Ektefa [3] examines the classifier execution, finding that neither C4.5 nor Support Vector Machines suit continuous multifarious situations. C4.5's approach is more precise and robust. Ching-Hao et al. [4] proposed a Co-getting ready framework to increase interruption I.D. by leveraging unlabeled data. Our new strategy has a better oversight rate than current methods do.

D.E. Denning has presented a Detecting and Checking framework to neutralize security breaches by identifying and handling anomalies in audit data. The suggested approach relies on profile-based representations concerning truthful models and estimations. Seth Ramalingam recommends cross-breeding feature assurance to handle the multidimensional dataset. The information gain inherent in the dataset is joined with the massive features through information gain-driven features discovery. The proposed technique has been shown to perform better when the attributes are combined. So, S. [6] This proposed technique has done away with an unproductive and boring part that slowed the depiction process.

According to Berchtold et al. [7], when working with a space of dimension d, pre-determining, estimating, and asking a course of action space for the nearest Neighbor issue can be done. Choosing the Voronoi diagram of the data centers may be a good start to planning future activity. To get an estimation of the Voronoi cells, the makers of this project recommend requesting. In high-dimensional spaces, this approach is practical for resolving the first nearest neighbors issue.

John Mchugh [8] suggested a test process using a combination of ruthless power examination to identify interruptions and to monitor unapproved usage, which depends on the employment of imprints and irregularity detection. Ravale et al. [9] recommended employing k-sketches and kernel components of support vector machines to perform interruption zone parts in the illustration system. To make data points stand out, the suggested system employs fewer properties for each point.

Wang Min and Gao Xiang [10] have presented unsupervised systems; they use large datasets as their preparation data and achieve reduced precision. This problem is solved by applying a semi-directed philosophy. The suggested methodology classifies data into the attack and regular classes. [12] Both recommended approaches show more excellent slanted screw and root mean squared error.

The proposal from Way T [12] uses a Markov process that mixes distinct irregularity and abuse attacks. The Semi-directed technique, which is composed of classifiers, is related in structure. Qiang Wang, Vasileios Megalooikonomou suggested fleecy validation, and Euclidean dissociations are employed to assess the proposed method.

The PSO (Particle Swarm Optimization) technique implemented by Holden [14] can consider specific attributes of apparent types. The proposed approach provides better precision when used with the direct standard.

The three authors, Zhang Fu, Marina Papatriantafilou, and Philippas Tsigas [17], suggest novel ways to diminish denial of service attacks. In essence, separate malicious traffic from trustworthy traffic and apply cheating systems linked with real traffic. PSO and SVM are brought together in the Ardjani framework to execute PSO and SVM. To ensure the accuracy of the claim, cross-endorsement of the first ten pages is done. The proposed method exhibits greater precision and will require more time to execute. Zhang Fu has drawn up a design for the bathroom sink tree. To counter repudiation of organization assaults, DDoS assaults don't merely target the target framework but all framework elements. Zhang et al. [15] advocated designing proactive systems to assemble the framework into sets. Each pack thinks approval should be granted at different meetings.

To limit the incidence of false alerts, Chien-Yi Chiu et al. [16] have developed a semi-managed technique to set up a warning channel with a high area rate. The structure in the proposed approach includes both semi-coordinated and managed learning elements.

KaiYan Feng [21] also described another neighborhood link known as nearest neighbors divorced from the central node. Two exhibits, An and B, are defined by a class L location point B, the close-by dormant kth-organize nearest Neighbor of An. To start, the general scores for each class are registered after their actual nearest neighbors, and their unapproachable nearest neighbors are learned. Class grades play a part in the question guides' placement, as the requests are dependent on this.

Vincenzo Gulisano et al. [9] have suggested a solution that allows for faster I.P. traffic screening by adding a prefix to differentiate anomalies associated with distributed denial of service attacks. The variable and static clock coast technique are suggested by Zhang Fu, Marina Papatrianta, et al., [10] to increase confirmation confidence. The standard technique can be based on the clock readout, depending on the speed and accuracy of the clock skim.

According to Li Jimin et al., [11], using a three-stage method to implement the SVM framework has allowed incredible speed and precision. The technique CLUS (Collection of data under the name of structure) was employed by Monowar H. Bhuyan [12] to implement a tree-based approach to locate groups without the need for named data. The structure we've presented is expected to offer superior outcomes for mixed kind data and numeric data. Carlos A. Catania and Carlos Garino [13] demonstrated the value of additional time usage in preparing framework data, which subsequently helps create the official pile.

III. PROPOSED ARCHITECTURE

The design plan The accompanying critical modules of the Intelligent Intrusion Detection Security System (IIDSS) are available. This group consists of -> It is made up of -> Preparation- Information that has already been pre-handled is accessible for use immediately, and attack-related information is retrieved from the KDD Cup dataset. Information from 23 special attacks is used to find disruption locations. Pre-preparing After utilizing the WEKA instrument, data pre-processing was concluded using the RemoveUseless() capacity to eliminate unnecessary attributes, reducing the number of attributes from 41 to 3 for execution improvement in the framework. Arrangement Simple K Nearest Neighbors Classification is the best classification for classifying attacks of this kind. Where were interruptions? Figure 2 shows the modules received.

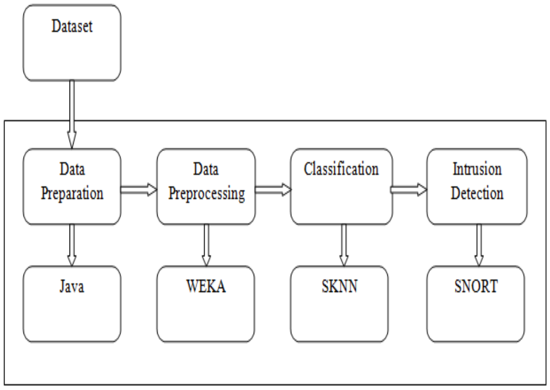

Figure 2: Proposed IIDS Architecture

IV. RESULTS AND OBSERVATION

The framework we currently use was conceived using the principles of both Hybrid PSO and C4.5. Here, the IDS framework is imbued with the concepts of SKNN Classifier brought to life in R. This research available in R, called "klaR," contains a bundle. The resulting data is of a sufficiently accurate kind. The results are outlined in Table 1.

Table 4.1: Results Comparison

Techniques	Sensitivity	Specificity	Accuracy	FAR
C4.5	87.57	83	91.24	1.45
SVM	81.92	63.29	88.27	3.01
C4.5+ACO	89.15	86.43	96.15	0.88
SVM+ACO	97.31	69.66	91.82	1.11
C4.5+PSO	93.40	89.88	96.37	1.83
SVM+PSO	91.50	71.10	92.59	2.96
EDADT	96.65	92.25	97.11	0.20
Proposed	99.81	99.90	99.62	0.01

V. CONCLUSION

The new intrusion detection system promises to be more effective than existing solutions. It has generated a 13.24% improvement in sentiment compared to C4.5, a 10.55% improvement compared to C4.5+ACO, and a 2.95% improvement compared to EDADT. While the existing framework appears to provide a low level of precision, the experimental result suggests that a higher level of precision is possible. The IDS will be better able to tell the difference between different types of attacks with additional future development, and the final count of attacks might go from 23 to 40. Because they can monitor network traffic, intrusion detection systems are great for spotting network packets. This study found that alarms are raised when packets' behavior deviates from normal. This process has been shown to increase productivity and decrease the number of fake alerts while also decreasing the stress on the management. The proposed snort rules base is compared to the existing signature patterns. The system was put through a thorough testing and comparison process, during which the suggested rules were found to be more accurate and efficient than existing snort rules. Future studies will utilize powerful data mining and machine learning methods to identify new suspicious attacks on large data sets.

VI. REFERENCES

[1] S. V. Lakshmi and T. E. Prabakaran, "Application of k-nearest neighbor classification method for intrusion detection in network data," in International Journal of Computer Applications, vol. 97, no. 7, 2014.

[2] S. O. Al-memory and F. S. Jassim Firas, "Evaluation of different data mining algorithms with KDD-CUP 99 dataset," in Journal of Babylon University/Pure and Applied Sciences, vol. 21, no. 8, pp. 2663–2681, 2013.

[3] N.S. Chandolikar and V.D. Nandavadekar, "Selection of relevant feature for intrusion attack classification by analyzing KDD-CUP 99," in MIT International Journal of Computer Science & Information Technology, vol. 2, no. 2, pp. 85–90, 2012.

[4] A. A. Olusola., A. O. Oladele, and D. O.Abosede, "Analysis of KDD'99 intrusion detection dataset for selection of relevance features," WCECS 2010, San Francisco, USA, 2010, vol. 1, pp. 20–22.

[5] K. C. Khor, C. Y. Ting, and S. P. Amnuaisuk, "From feature selection to building of Bayesian classifiers: a network intrusion detection perspective," in American Journal of applied sciences, vol. 6, no. 11, p. 1948, 2009.

[6] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," 2003 SIAM International Conference on Data Mining, 2003, pp. 25–36.

[7] A. O. Adetunmbi, S. O. Falaki, O. S. Adewale and B. K. Alese, "Network intrusion detection based on rough set and k-nearest neighbor," in International Journal of Computing and ICT Research, vol.2, no.1, pp. 60–66, 2008.

[8] S. Chebrolu, A. Abraham and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," in Computers & Security, vol. 24, no. 4, pp. 295–307, 2005.

[9] S. Mukkamala, A. Sung, "Significant feature selection using intelligent computational techniques for intrusion detection," Advanced Methods for Knowledge Discovery from Complex Data, 2005, pp. 285–306.

[10] S. J. Horng, M. Y. Su, Y. H. Chen, T. W. Kao, R. J. Chen, J. L. Lai, and C. D. Perkasa, "A novel intrusion detection system based on hierarchical clustering and support vector machines," in Expert systems with Applications, vol. 38, no. 1, pp. 306–313, 2011.

[11] F. Amiri, M. R. Yousefi, C. Lucas, A. Shakery and N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," in Journal of Network and Computer Applications, vol. 34, no. 4, pp. 1184–1199, 2011.

[12] M. S. Roulston, "Estimating the errors on measured entropy and mutual information," in Physica D: Nonlinear Phenomena, vol. 125, no. 3, pp. 285–294, 1999.

[13] H. Peng, F. Long and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and minredundancy," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 8, pp. 1226–1238, 2005.

[14] F. Rossi, A. Lendasse, D. Franc¸ois, V. Wertz, M. Verleysen, "Mutual information for selecting relevant variables in nonlinear spectrometric modeling," in Chemometrics and intelligent laboratory systems, vol. 80, no. 2, pp. 215–226, 2006.

[15] A. Kraskov, H. St¨ogbauer, and P. Grassberger, "Estimating mutual information," in Physical Review E, vol. 69, no. 6, p. 66138, 2004.

[16] A. Gouveia and M. Correia, "Feature set tuning in statistical learning network intrusion detection," 2016 IEEE 15th International Symposium on Network Computing and Applications, 2016, pp. 68–75.

[17] S. O. Al-memory and F. S. Jassim, "Evaluation of different data mining algorithms with KDD-CUP 99 dataset," in Journal of Babylon University/Pure and Applied Sciences, vol. 21, no. 8, pp. 2663–2681, 2013.

[18] D. Deepika and V. Richhariya, "Intrusion detection with kNN classification and DS-Theory," in International Journal of Computer Science and Information Technology and Security, vol. 2, no.2, pp. 274–281, 2012.

[19] B. Subba, S. Biswas, and S. Karmakar, "Intrusion detection systems using linear discriminant analysis and logistic regression," 2015 Annual IEEE India Conference (INDICON), New Delhi, 2015, pp. 1-6.

[20] S. Devaraju and S. Ramakrishnan, "Performance comparison for intrusion detection system using a neural network with KDD dataset,"ICTACT Journal on Soft Computing, vol. 4, no. 3, 2014.

[21] A. A. Olusola, A. S. Oladele, and D. O. Abosede, "Analysis of KDD'99 intrusion detection dataset for selection of relevance features," world Congress on Engineering and Computer Science, San Francisco, USA, 2010, vol. 1, pp. 20–22.

[22] R. Chitrakar and C. Huang, "Anomaly based intrusion detection using hybrid learning approach of combining k-medoids clustering and Na¨ıve Bayes classification," 2012 8th International Conference on Wireless Communications, Networking, and Mobile Computing (WiCOM), Shanghai, 2012, pp. 1-5.

[23] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," IEEE Transactions on Computers, vol. 65, no. 10, 2014, pp. 2986-2998.