International Journal of Advancement in Electronics and Computer Engineering (IJAECE)
Volume 11, Issue 11, November. 2022, pp.113-117, ISSN 2278 -1412

Optimization Accuracy of Distributed Denial of Service Attacks in SDN based on Boosting Algorithm

[1]Sushrita Mishra, [2]Dr. Kalpana Rai, [3]Asst. Prof. Sneha Soni
Department of Computer Science and Engineering,
Sagar Institute of Research and Technology Excellence, Bhopal, India

Abstract- *Software-Defined Networking (SDN) is a contemporary and growing technology in the field of networking. It has the benefit of decoupling the Data plane and the control plane, thus enabling automated provisioning. The SDN offers numerous benefits, such as dynamic programmability, high bandwidth, and cost-effectiveness; it improves network performance than the existing traditional networks. SDN environment is against high-rate DDoS flooding attacks effectively through the use of a two-level security mechanism. Level-I and Level-II were implemented using an entropy-based mechanism and the artificial neural (AN) based boosting technique, respectively. First, traffic instances pass through the Level-I module. If a packet is suspected to be an attack, based on the entropy value calculated, a temporary alert is generated to hold the particular flow. The entropy value is calculated using the randomness of IP addresses in a particular time window. Next, the traffic instances pass through the Level-II module. Essential features extracted from the flow information to improve the accuracy rate included the precision, recall and loss.*

Keywords- ***SDN, DDoS Attack, Accuracy, AN, Boosting Algorithm***

I. INTRODUCTION

The Internet has led to tremendous innovations in technology. Corporations are compelled to switch to flexible and modern network technologies for business processing. Computer backbone networks generally consist of numerous switches and routers controlled by a network administrator. Network administrators configure network policies and update them regularly to adapt to dynamic changes in network traffic. Traditional network architecture has difficulty in handling advanced and dynamic changes in configuration, owing to a lack of automated update mechanisms which impacts flexibility and innovation in networking infrastructure. Security policies can only be imposed when each network device is configured individually by vendor-specific commands, thereby increasing configuration complexity and curbing dynamic changes being made to the network [1]. To overcome problems in traditional networks and improve the quality of service, new SDN paradigm has emerged. SDN is a powerful technology that addresses problems through innovations in networking development and research. A SDN is a three-layered structure consisting of a data plane, a control plane and an application plane [2, 3]. The separation of the data and control planes in the SDN is accomplished by providing a programming interface between the network device and SDN controller through an application programming interface (API). The controller in the control plane judges how best to manage network traffic and network devices in the data plane forward traffic in line with decisions made by the controller [4, 5]. The separation provides a high level of flexibility and scalability, apart from permitting the network administrator to make dynamic changes in the network. Hence, the controller constitutes the core of the SDN and demands maximum security. The data plane comprises network switching elements with no control functionalities and forward traffic in accordance with the controller's decisions. Network elements communicate with the controller via an interface called a southbound interface (SBI) or control-data plane interface [6]. The application plane constitutes security and business applications as well as virtual network and memory management functions. The controller communicates with application plane applications through a northbound interface (NBI). Fig. 1 depicts the architecture of SDN.

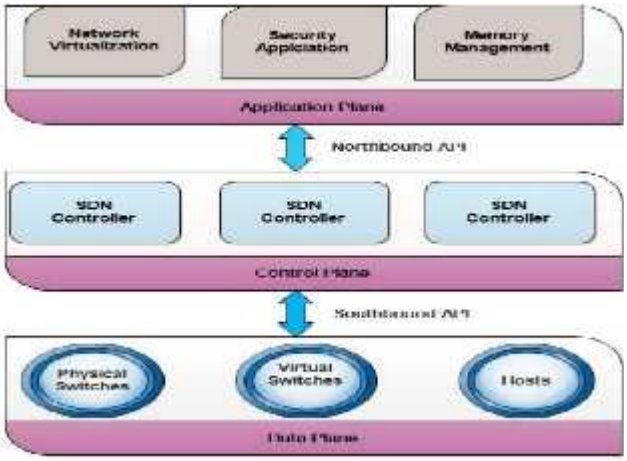


Fig. 1: SDN architecture

II. BENEFITS OF SDN

The development of technologies like big data, the cloud and virtualization has put pressure on traditional networks typically characterized by a lack of innovation, slow development and production delays. A networking environment that is optimized and automated for dynamic changes makes the SDN powerful, leading to rapid innovation in development and production. The SDN deals with problems in traditional networks by merging the centralized controller and data plane switches into a single unified platform.

A clear advantage the SDN offers is the ability to quickly process entire requests from different devices using a programming interface. The software-based controller lets users and administrators manage the traffic flow efficiently with an abstract view of the network. Network administrators increase bandwidth and other resources as needed while simultaneously investing little on additional physical infrastructure.

Centralized intelligence in the SDN transforms the networking function, making it dynamic and powerful. The SDN offers integration with the public cloud, an abstract view of the entire network infrastructure, the management of virtual and physical devices from the centralized controller and low operational costs. The SDN helps to reduce the overall operating costs by automating and centralizing the administrative process [7, 8]. An added advantage of the SDN is its efficient control of data traffic, which automatically enhances the quality of service (QoS) for multimedia transmissions and Voice over IP (VoIP) [9]. Technologies like cloud computing, big data and virtualization increasingly demand dynamic and flexible networks. Such demands have resulted in IT enterprises and the corporate sector switching to SDN services for superior performance, innovations, reduced costs and complexity [10].

III. CHALLENGES & ISSUES

3.1 Challenges

The SDN facilitates powerful and dynamic networking for IT enterprises and communication service providers. There are, however, challenges to be addressed in terms of availability, reliability, scalability and controller placement, as well as security issues like denial-of-service and man in-the-middle attacks, along with vulnerability scans [11]. The SDN controller is prone to single-point failures and its centralization makes it an easy target for attacks. With the controller compromised, network manipulation attacks are initiated. Further, compromised data plane devices cause a series of traffic diversions, as well as side-channel and traffic-sniffing attacks. Given that the controller processes every new entry, the focus of the attacks is on disruption. Controller failure results in a network collapse and service cuts for a long while, which degrades services to IT enterprises.

Data plane devices and the control plane controller work independently, communicating information only through the API. When the number of devices in the data plane increases, communication between the single controller and multiple devices hits a bottleneck and leads to problems with scalability. Using a proper and efficient mechanism to protect the SDN controller helps ensure its security, availability, reliability and scalability. Our research essentially focuses on security issues in the SDN data plane and control plane [12].

3.2 Issue

Providing and managing security in a computer network is complex, with the network administrator having to provide security to safeguard it from internal and external intruders. A 2019 survey of cyberattacks revealed that more than 64%, 62%, 59% and 51% of all companies faced web-based, phishing, botnet and DDoS attacks, respectively [13]. Network intrusion is intended to disrupt the target's system resources by stealing confidential information, disabling its functionalities and launching miscellaneous attacks. The threat looms large for large-scale organizations just as much as it does for medium and small-scale ones, because the latter are often unable to afford high-level security measures. Often, it results in attackers focusing on vulnerable medium and small-scale organizations. The most common threats to networks are from malware, SQL injection, phishing, botnet, cross-site scripting, DoS and DDoS attacks [14]. Malware attacks occur across devices and operating systems. They are fairly common attacks where malicious software is developed and installed to gain access to a victim's system without their knowledge. The attacks seek access to the victim's personal information in the form of credentials and confidential information; damage the system's resources and gain complete access for financial gain. Major malware attacks include spyware, viruses and ransomware [15, 16].

IV. PROPOSED METHODOLOGY

Techniques or models based on the principles of learning are called as NN. NN are categorized in terms of three basic entities which include the core processing element called as neuron, the interconnection structure and the learning algorithm. Performance of an ANN is defined by its basic architecture which includes various parameters like number of hidden layers in an ANN, the neuron (node) count in each of these layers, transfer function used at each node, weights and parameters of the training algorithm used including their settings too. A general model for software cost estimation based on ANN is shown below:

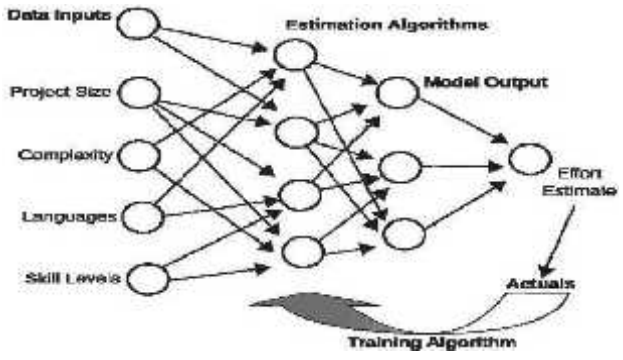


Fig. 2: Neural Network

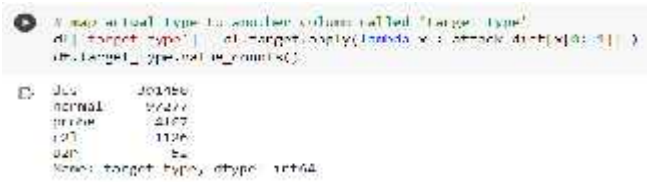
Step 1: Data set collection

According to dataset attribute information

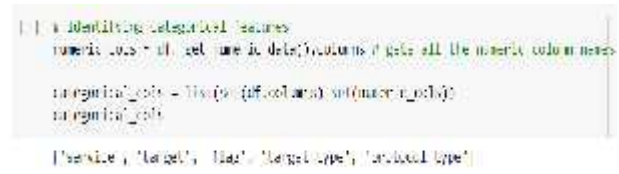
- target column 'Normal' represents Good Connection
- Bad connection attack types are

- DoS(Denial of Service)
- User to Root(U2R)
- Remote to Local(R2L)
- Probe

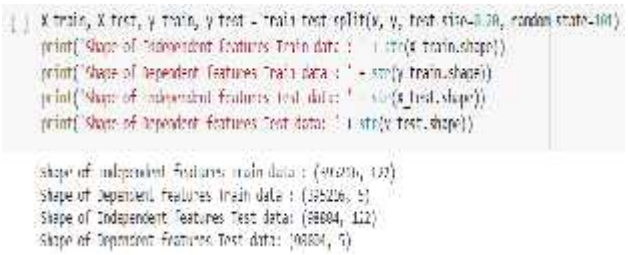
Files used kddcup.data_10_percent.gz, kddcup.names, training_attack_types



Step 2: Categorical Features Exploration and Analysis



Step 3: Split Data into training and testing purpose into 80,20 ratio



Step 4: Defining NN Proposed ANN

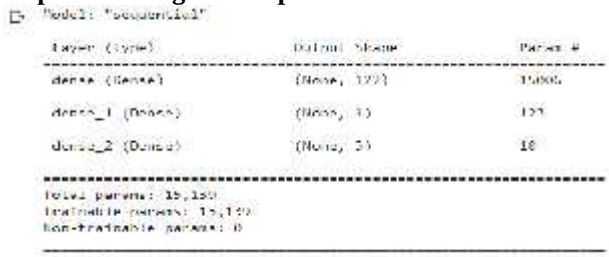


Table 1: Training Hyper Parameters

Layers	Dense
Model	Sequential
Neurons	128
Activation function	Relu
Kernal Initializer	Random Uniform
Output Activation Function	Softmax
Training data	80%
Testing data	20%
Loss	Categorical Cross Entropy
Optimizer	Adam
Epochs	50

V. RESULT ANALYSIS

Generally, the performance of a classification model is evaluated in terms of accuracy, sensitivity and specificity to calculate which the values of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) need to be considered. A good machine learning model requires high accuracy and low false alarm rates. A confusion matrix is used to determine these parameters. In the confusion matrix, true positive is the number of normal records correctly identified as normal records; false positive is the number of normal records incorrectly identified as attacks; true negative is the number of attack records correctly identified as attacks and false negative is the number of attack records incorrectly identified as normal records.

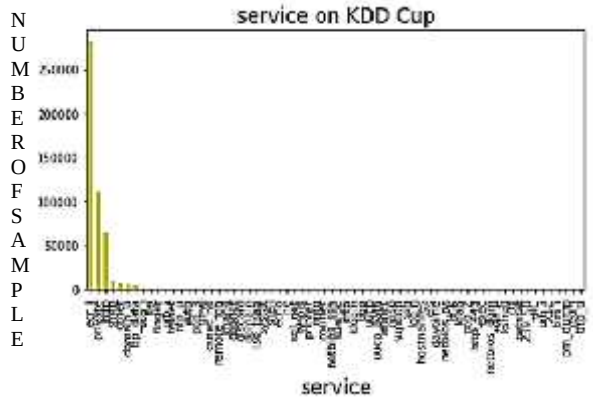


Fig. 3: Different types of Service on KDD Cup

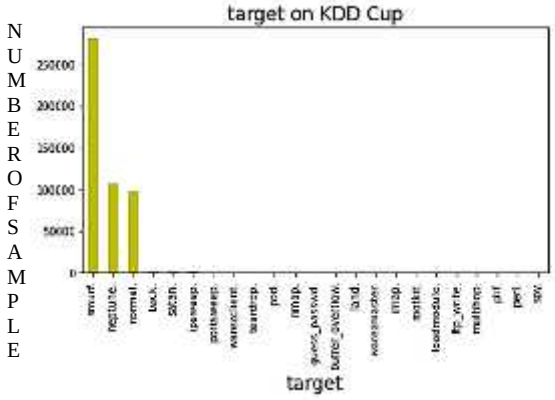


Fig. 4: Different types of target on KDD Cup

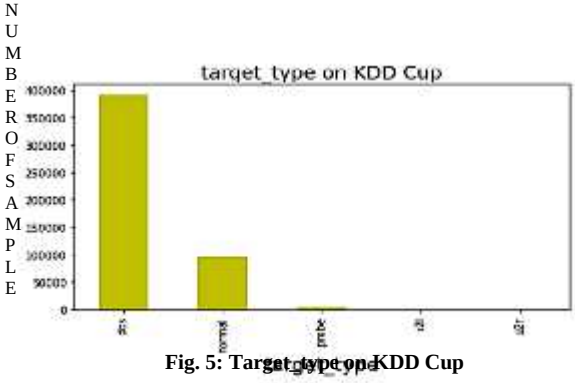


Fig. 5: Target_type on KDD Cup

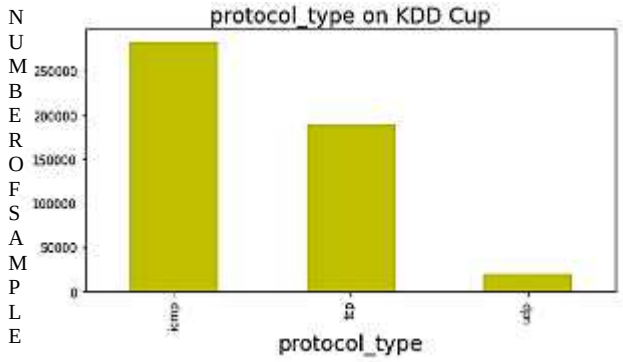


Fig. 6: Protocol_type on KDD Cup

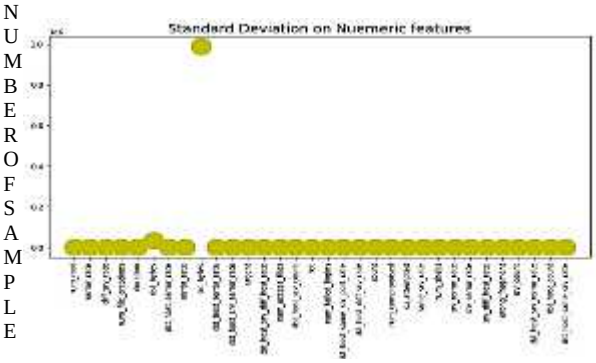


Fig. 7: Standard Deviation on Numerical Features

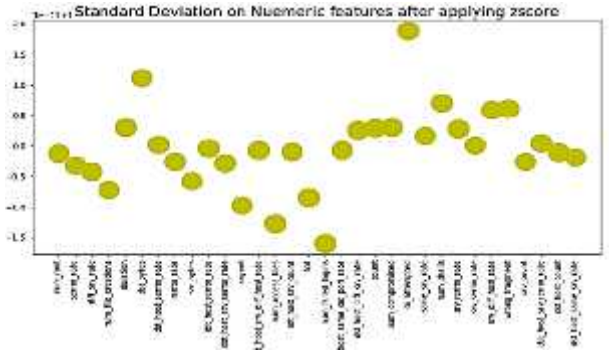


Fig. 8: Standard Deviation on Numerical Features after applying z- score

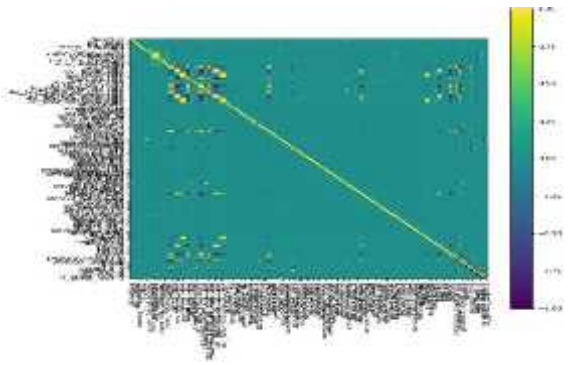


Fig9 : Exploratory Data Analysis

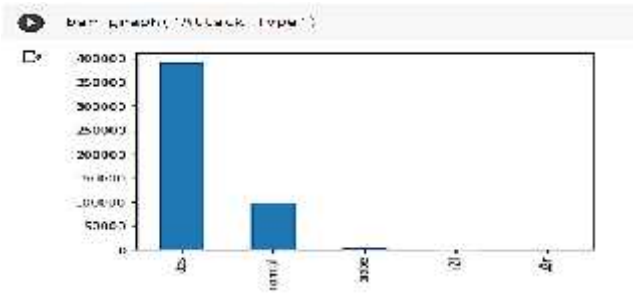


Fig. 10: Different Attack Types

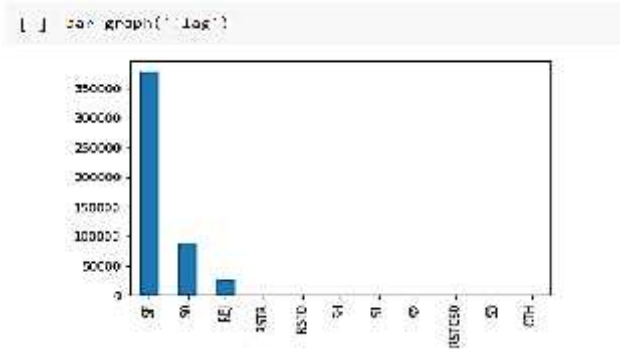


Fig. 11: Different Flag Types

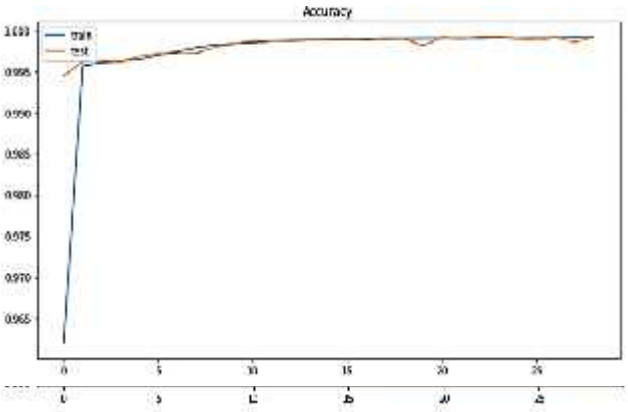


Fig. 13: Accuracy for Test and Training

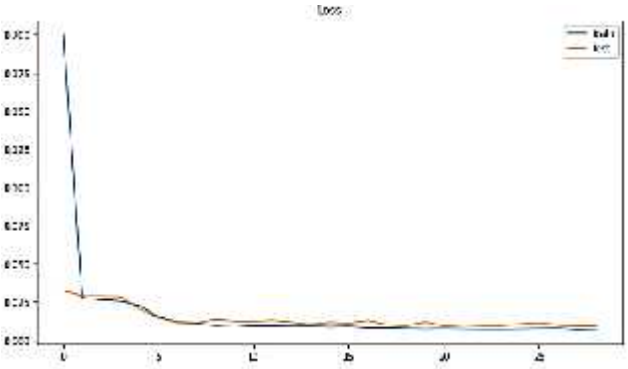


Fig. 13: Loss for Test and Training

Table II: comparison Result

Algorithms	Precession	Recall	F1_Score	Accuracy	Loss
Decision Tree	92	9	95	97	3
SVM	100	72	84	76	33
Proposed ANN	100	100	100	99	0.001

VI. CONCLUSION

Normal detection processes, like IP address monitoring are unhelpful in detecting LR-DDoS attacks. Since normal features like the port number, source and destination address are just not adequate enough to detect attacks, we used four essential features: the relative distribution of packet intervals, the number of packets, the duration of the flow and the relative distribution of match bytes from the SDN flow. The use of advanced ANN models will enable automatic feature extraction and thus could improve the performance of the system as a whole.

REFERENCES-

[1] K. Muthamil Sudar, M. Beulah and P. Deepalakshmi, "Detection of Distributed Denial of Service Attacks in SDN using Machine learning techniques", International Conference on Computer Communication and Informatics (ICCCI), Jan. 27 – 29, 2021, Coimbatore, INDIA.

[2] Muthamil Sudar, K., & Deepalakshmi, P. (2020). A two level security mechanism to detect a DDoS flooding attack in software-defined networks using entropy-based and C4. 5 technique. Journal of High Speed Networks, (Preprint), 1- 22.

[3] Dong, S., & Sarem, M. (2019). DDoS Attack Detection Method Based on Improved KNN With the Degree of DDoS Attack in Software-Defined Networks. IEEE Access, 8, 5039-5048.

[4] Dong, S., Abbas, K., & Jain, R. (2019). A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments. IEEE Access, 7, 80813-80828.

[5] Gu, Y., Li, K., Guo, Z., & Wang, Y. (2019). Semisupervised K-means DDoS detection method using hybrid feature selection algorithm. IEEE Access, 7, 64351- 64365.

[6] A. Raghavan, F. D. Troia, and M. Stamp, ``Hidden Markov models with random restarts versus boosting for malware detection," *J. Comput. Virol. Hacking Techn.*, vol. 15, no. 2, pp. 97107, Jun. 2019.

[7] T. Young, D. Hazarika, S. Poria, and E. Cambria, ``Recent trends in deep learning based natural language processing [review article]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 5575, Aug. 2018.

[8] X. Lei and Y. Xie, ``Improved XGBoost model based on genetic algorithm for hypertension recipe recognition," *Comput. Sci*, vol. 45, pp. 476481, 2018.

[9] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, ``Deep learning for visual understanding: A review," Neurocomputing, vol. 187, pp. 2748, Apr. 2016.

[10] Abduvaliyev, A., Pathan, A.-S. K., Zhou, J., Roman, R., and Wong, W.-C. "On the Vital areas of Intrusion Detection Systems in Wireless Sensor Networks", IEEE Communications Surveys & Tutorials, Vol. 15, Issue 3, pp. no. 1223–1237, 2015.

[11] Abubakar, A. I., Chiroma, H., Muaz, S. A., and Ila, L. B. "A Review of the Advances in Cyber Security Benchmark Datasets for Evaluating Data-driven based Intrusion Detection Systems", Procedia Computer Science, Vol. 62, pp. no. 221–227, 2015.[12] Bay, S. D., Kibler, D., Pazzani, M. J., and Smyth, P. (2015), "The UCI KDD archive of Large Data Sets for Data Mining Research and Experimentation", ACM SIGKDD Explorations Newsletter, Vol. 2, Issue 2, pp. no. 81–85, 2015.

[13] Aburomman, A. A. and Reaz, M. B. I. "A novel SVM-kNN-PSO ensemble method for Intrusion Detection System. Applied Soft Computing", Vol. 38, pp. no. 360–372, 2015.

[14] Pedro Casas, JohanMazel and Philippe Owezarski "Unsupervised Network Intrusion Detection Systems: Detecting the Unknown without Knowledge", Elsevier Computer Communications, Vol. 35, Issue 7, pp. no. 772 – 783, 2012.

[15] Carlos A. Catania, Facundo Bromberg and Carlos García Garino "An Autonomous Labeling Approach to Support Vector Machines Algorithms for Network Traffic Anomaly Detection", Elsevier Expert Systems with Applications, Vol. 39, Issue 2, pp. no. 1822–1829, 2012.

[16] Xie, B & Zhang, Q, "Application-layer anomaly detection based on application-layer protocols' keywords", Computer Science and Network Technology (ICCSNT), 2nd International Conference on, pp. 2131-2135, 2012.