© 2015 IJSRSET | Volume 1 | Issue 6 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Detection of Anomalous Behavior for Real Time Wide Area Network Traffic Using Wireshark

Shivendu Dubey, Neha Tripathi
Gyan Ganga Institute of Technology & Science, Jabalpur, Madhya Pradesh, India

ABSTRACT

Interruption identification is a compelling methodology of managing issues in the territory of system security. Quick improvement in innovation has raised the requirement for a successful interruption discovery framework as the customary interruption identification technique can't go up against recently propelled interruptions. As most IDS attempt to perform their assignment continuously however their execution upsets as they experience distinctive level of examination or their response to confine the harm of a few interruptions by ending the system association, an ongoing is not generally accomplished. With expanding number of information being transmitted step by step starting with one system then onto the next, the framework needs to distinguish interruption in such huge datasets viably and in an auspicious way. In this manner, the utilization of information mining and machine learning methodologies would be successful to recognize such abnormal get to or assaults. Additionally, enhancing its execution and precision has been one of the significant tries in the examination of system security today. In this exploration, we have actualized an interruption discovery framework (IDS) in light of exception ID managing TCP header data utilizing WIRESHARK.

Keywords : IDS, TCP, Wireshark, FTP, Hyper-Media System, SMTP, IDS, TTL, NetSTAT, DIDS, ICMP

I. INTRODUCTION

Intrusion identification is a system security instrument to shield the PC net-work framework from intrusion or assaults. Headway in system advancements has given an opening to programmers and gatecrashers to discover unapproved approaches to go into another framework. Subsequently, as advances advance, there is additionally a danger of new dangers existing with them. Subsequently, when another sort of intrusion develops, an intrusion discovery framework (IDS) should have the capacity to act adequately and in an auspicious way to stay away from dangerous impacts. In today's setting, the greatest challenge could be managing 'huge information', i.e., a gigantic volume of system movement information that gets gathered powerfully in the system correspondences [1]. Along these lines, interruption location has been one of the center territories of PC security whose objective is to recognize these vindictive exercises in system activity and, vitally to shield the assets from the danger. Most IDS attempt to perform their undertaking progressively yet they need because of different reasons. The circumstances like level of buttcentricys are and calculation it needs to experience, the constant execution is not generally possible.

In our approach, we have tried to inspect the TCP headers information from the Transmission Control Protocol/Internet Protocol (TCP/IP) packets and detect attacks as an outlier based on the analysis of these header information. We can find the majority of TCP/IP suite of protocols being used for internet data communi cations applications which include the World Wide Web hyper-media system that uses HTTP (Hyper Text Transfer Protocol). Other examples are common network protocols such as FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Pro- tocol) and Telnet. The common use of TCP means that it is likely to be exploited for misuse and various forms of attacks. Thus, there is a possibility that malicious behaviour can be executed through the TCP/IP protocols without being

blocked or even noticing by firewalls or detection by IDS because most commonly used IDS do not recognize new variations of attacks. The measure of detection is the deviations of anomaly TCP header data from the normal TCP header data which allows for high speed network detection since extracting a TCP/IP header information can be performed in minimal time. The useful information that can be used from TCP headers are IP Length, TCP flags, TCP window size, checksum and time-to-live (TTL). Many researchers have highlighted the fact that anomaly connections have different patterns of TCP flags from normal ones.

II. METHODS AND MATERIAL

Related Work

Intrusion Detection Model

In 1986, the first intrusion detection model was proposed by Denning which can be discuss in The model was not specific to any system and inputs, but it referred to a reference value for inputs using system and machines. The model generates a number of contours and monitors the contours change based on audit log data of the host system and hence, finds intrusions in the system [1].

Intrusion Detection System can be categorized into two groups based on its deployment.

- Host Based IDS
- Network Based IDS

Host based systems mostly utilizes the system logs in order to monitor the attacks in the individual hosts. While network based systems is about building enterprise level security systems focusing on huge network traffics and analyzing the attacks from the outside. Host based system are small and easy to deploy and thus, preferable for personal use while network based systems may require higher end of advanced hardware and softwares in order to protect the whole subnet. Network based IDS can be further divided into network monitoring systems and composite systems that monitors both hosts and the surrounding network.

Network Security Monitor (NSM) [12], Bro [10], Network Flight Recorder (NFR) [11] and Network Statistics (NetSTAT) [9] are the example of available strict network based systems. NSM was a system designed to monitor the traffic between hosts. Bro acts as a high-speed passive network monitor that filters traffic for certain applications. NFR was a tool for filtering the network data while Net- STAT offered customization and filtering of network events [11]. Emerald, Grids and Distributed Intrusion Detection System (DIDS) are such examples of system that monitors both hosts and network. Emerald was able to detect intrusion in largely distributed networks that would respond based on local targets and manage its monitors to form an analysis hierarchy of network wide threats. Grids allow easy viewing of attacks via graph by collecting results from both host and network based components. DIDS is like an extension of NSM, that utilizes data from both host auditing systems and LAN traffic to detect intrusions [12].

In order to build effective intrusion detection, there are different datasets available for research work and previously many competitions were and are still being held. The MIT Lincoln Labs developed Darpa ‘98 and ‘99 dataset [5] and its modified version, KDD ‘99 dataset [3] which is freely available for the research community to work on intrusion detection problems. Darpa dataset and KDD dataset both are very large dataset consisting of US air force local network data with a variety of simulated attacks. KDD dataset consists of 41 features, defined for each connec- tion samples which are further divided into four categories namely features of TCP protocol, content features, time-based and host-based traffic features. It was no- ticed that NSL-KDD dataset [5] was developed to overcome the problems related to KDD 99 dataset. It was done by reducing the redundant problem in the dataset. It significantly reduced the size of the dataset that made the data evaluation and validation of the learning algorithm much easier and convenient. The removal of redundant records helped researchers to use dimension reduction techniques like Rough Set Theory based classifiers which make their work unbiased towards both the frequent and infrequent records, and helped them to get a more consistent result as claimed by Suthaharan et al. [1]. The other datasets that are known for testing IDS are ECML-PKDD HTTP 2007 [6] and CSIC HTTP 2010 [7].

III. RESULTS AND DISCUSSION

Proposed Work and Results

When capturing packets from a network interface, wireshark captures all of the packets coming and going over the network. Wireshark provides capture and display filters which allow you to capture only the packets you are interested and display allows you to specify which packets are shown in GUI. We use port numbers also that can be used to capture packets that are destined for certain applications. Like UDP had port no 53. HTTP has 80 and so on [4]. For analysis we use a term Protocol dissector. A. Protocols coming under capturing of live network are

1. TCP/IP
2. HTTP
3. ARP
4. ICMP

TCP/IP and HTTP Protocol - It is most widely used network protocol. HTTP is a server/client based protocol used to transfer web pages on network. TCP/IP is a stack of protocols having different protocols on both layer 3 and 4. It is a layer 4 protocol and provides bi directional communication. IP is a layer 3 protocol and provides addressing system that allows communication on network. It is a connectionless protocol [3].

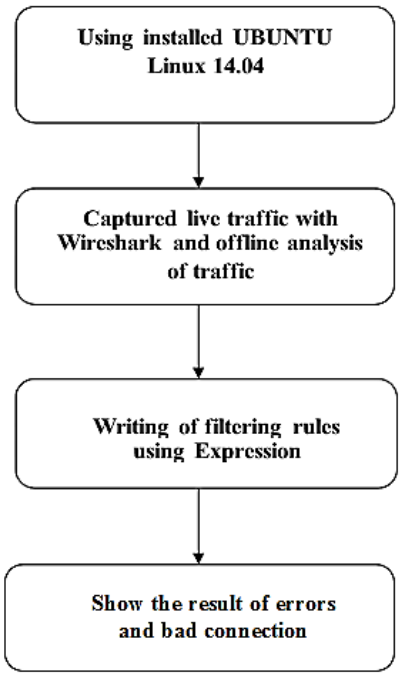

Figure 3.1: Implementation setup Diagram

Time Sequence graph (tcp-trace) graph displayed is as shown in Figure 3.2 below.

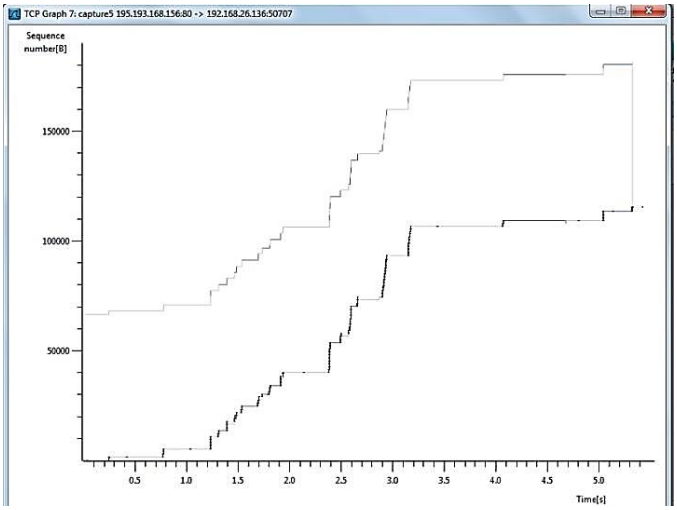

Figure 3.2 : Time Sequence graph (tcp-trace) graph

I/O Graph : - Most important facility provided by Wireshark is to draw I/O graph of the captured packets. At a time we can draw graph of five protocols of different colours with different tick interval and pixels per tick on X-axis and units and scale on Y-axis. Styles can also be changed instead of lines shown you can select impulse, Fbar, dot from the drop down and you get a different look of the graph. A line graph of TCP, HTTP, and ARP protocol is shown to you in Figure 3.3.

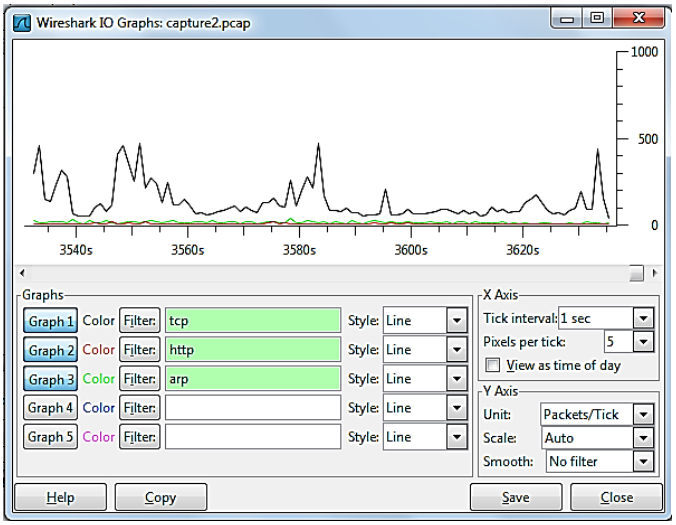

Figure 3.3 : Comparison Graph of Captured Trace TCP, HTTP and ARP Protocol

Network traffic from a live network is shown by taking various traces and monitoring and analysis is done on

that captured files and then statistics is built. Detailed analysis and summary as well as conversations between two end points are shown. One interesting option which Wire shark give is objects which we captured or say user who are on the network using whatever sites can be listed in this object list. Graphs of captured files are shown and other attractive features are shown which make Wireshark a great tool for network analysis. The Output graph generated through captured packets provides details of network dynamics and insight into the problems that lead to network slowness, network performance etc.

IV. CONCLUSION

In this paper Applicable components should be extricated and chose by envisioning their examples and lessening their dimensionality. Machine Learning methodology was acquainted in execution of IDS with reduction the level of human connection and expands the powerful of IDS. Numerous IDS still does not have the capacity to identify a wide range of new assaults in the system, so scientists are slanted towards displaying the typical occurrences to expand their framework viability.

Our methodology beats the downsides of one connected with the principle based methodologies and is effective. We have talked about the adequacy of our work on premise on bunching examination, execution measurements, ROC bend and accuracy review test. The yield of the prepared results in real time traffic was likewise a promising sign for our model to be viable in some other circumstances. Accordingly, our work gives a functional answer for development of better IDS taking into account 'Exception Detection technique using WIRESHARK'. Therefore, with our outcomes and examination, we can say that, we have investigated the TCP header data that permits to manage irregularity location in quick approaching movement progressively with lower false positive discovery rate on constant information activity.

V. REFERENCES

[1] S. Suthaharan and T. Panchagnula, "Relevance feature selection with data cleaning for intrusion detection system," in 2012 Proceedings of IEEE South- eastcon, 2012, pp. 1-6.

[2] X. Zhang, L. Jia, H. Shi, Z. Tang, and X. Wang, "The Application of Ma- chine Learning Methods to Intrusion Detection," in 2012 Spring Congress on Engineering and Technology (S-CET), 2012, pp. 1-4.

[3] F. Gharibian and A. A. Ghorbani, "Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection," in Fifth Annual Conference on Communication Networks and Services Research, (CNSR), 2007, pp. 350-358.

[4] I. H. Witten and E. Frank, "Data mining practical machine learning tools and techniques," San Francisco: Morgan Kalfman, 2005.

[5] R. Groth, "Data Mining: Building Competitive Advantage," USA: Prentice Hall, 2000.

[6] H. Sarvari and M. M. Keikha, "Improving the accuracy of intrusion detection systems by using the combination of machine learning approaches," in 2010 International Conference of Soft Computing and Pattern Recognition (SoCPaR), 2010, pp. 334-337.

[7] H. T. Nguyen and K. Franke, "Adaptive Intrusion Detection System via online machine learning," in 2012 12th International Conference on Hybrid Intelligent Systems (HIS), 2012, pp. 271-277.

[8] T. S. Chou, J. Fan, S. Fan, and K. Makki, "Ensemble of machine learning al- gorithms for intrusion detection," in IEEE International Conference on Systems, Man and Cybernetics, (SMC) 2009, pp. 3976-3980.

[9] X. Liao, L. Ding, and Y. Wang, "Secure Machine Learning, a Brief Overview," in 2011 5th International Conference on Secure Software Integration Reliability Improvement Companion (SSIRI-C), 2011, pp. 26-29.

[10] J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations As Performed by Lincoln Laboratory," in ACM Transactions Information System Security, vol. 3, no. 4, Nov. 2000, pp. 262-294.

[11] D. Kershaw, Q. Gao, and H. Wang, "Anomaly-Based Network Intrusion De- tection Using Outlier Subspace Analysis: A Case Study," in Advances in Artifi- cial Intelligence, C. Butz and P. Lingras, Eds. Springer Berlin Heidelberg, 2011, pp. 234-239.

[12] W. Da and H. S. Ting,"Distributed Intrusion Detection Based on Outlier Min- ing," in Proceedings of the 2012 International Conference on Communication, Electronics and Automation Engineering, G. Yang, Ed. Springer Berlin Heidel- berg, 2013, pp. 343-348.

[13] N. Devarakonda, S. Pamidi, V. V. Kumari, and A. Govardhan,"Outliers De- tection as Network Intrusion Detection System Using Multi Layered Frame- work," in Advances in Computer Science and Information Technology, N. Meghanathan, B. K. Kaushik, and D. Nagamalai, Eds. Springer Berlin Heidel- berg, 2011, pp. 101-111.

[14] J. Zhang and M. Zulkernine,"Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection," in IEEE International Conference on Communications, (ICC) 2006, vol. 5, pp. 2388-2393.

[15] V. Pareek, A. Mishra, A. Sharma, R. Chauhan, and S. Bansal,"A Deviation Based Outlier Intrusion Detection System," in Recent Trends in Network Se- curity and Applications, N. Meghanathan, S. Boumerdassi, N. Chaki, and D. Nagamalai, Eds. Springer Berlin Heidelberg, 2010, pp. 395-401.