Network Security Analysis Using Big Data Technology

Yogeshwar Rao Bachupally
Department of Computer Science
NC A&T State University
Greensboro, NC, USA
ybachupa@aggies.ncat.edu

Xiaohong Yuan
Department of Computer Science
NC A&T State University
Greensboro, NC, USA
xhyuan@ncat.edu

Kaushik Roy
Department of Computer Science
NC A&T State University
Greensboro, NC, USA
kroy@ncat.edu

Abstract—**With the evolution of networks, threats or attacks with the intention of disrupting service or stealing confidential data are increasing tremendously. Networks have to be monitored constantly and protected against attacks. In this paper, a new method to analyze network traffic using Big Data techniques is introduced. This approach detects anomalous activities being carried out and malicious data being transmitted over the networks through processing and loading traffic data into Hive database in Hadoop Distributed File System (HDFS) environment, and analyzing the data using Hive queries. The results of using this method to detect attacks on the sample dataset are also presented.**

Keywords—Network security analysis, Big Data, Hadoop Distributed File System

I. Introduction

Threats to the integrity of a network produce challenges for the integrity and operational capability as well as the cost involved with operating and maintaining it. Network security [1] involves measures to detect, deter, prevent and correct security violations that involve the transmission of information. Three concepts that embody the fundamental security objectives are Confidentiality, Integrity and Availability (CIA). A network is said to be secure if it maintains and preserves CIA. To make sure if a network maintains CIA, the network traffic has to be monitored constantly against different types of attacks using signature based detection as well anomaly based detection methods. The first step to detect/prevent attacks is to analyze the connections in the network, data being transmitted over the network and the type of requests being made. If the network is of small size it would not be a difficult task to constantly monitor and analyze the network but in case of large networks it would be very difficult to carry out the analysis and get the metrics related to connections, requests and type of data being transmitted so as to protect the network from zero-day attacks.

In a world, where there are already billions of devices connected together and more new devices getting connected every day, it is becoming extremely difficult to analyze the network connections, data being transmitted and protect it with the existing conventional detection and analysis techniques. For some of the networks, the size of the log data grows to size of TBs which require a powerful computation approach to carry out the network analysis.

In this paper we present the technique of analyzing the network traffic data and detect anomalous connections to the network using big data and Hadoop Distributed File System (HDFS) [3, 4]. Hadoop utilizes the concept of parallel computing in which the computation or application that does the analysis is transmitted to the data instead of transmitting data to memory where the application runs. The time taken to complete the computation or analysis directly depends on the number of machines that do parallel computing and aggregate the results obtained from each computation.

The analysis of network security is carried upon the data captured related to the connections, type of requests made, data transferred over the network using the network capture tool Wireshark. The captured data is in PCAP format and consists of sample log data from the first day of the 2014 national collegiate cyber defense competition (NCCDC) provided by PREDICT cyber security dataset provider [20]. The log data is formatted and data related to window size, TCP flags etc., is extracted from the log data to detect intrusions, malicious data transmitted, anomalous connections etc. The extracted data is exported to .csv format and is uploaded to HDFS environment. The extracted data is then imported to Hive Database that provides a structured query language dialect called Hive Query Language (HQL) [13, 14]. HQL hides the complexity of MapReduce programming [2, 5, 6] in HDFS. Queries constructed to extract the metrics related to the malicious content, attacks detected and the results of the analysis are presented in this paper.

The rest of the paper is organized as follows. In section II, the related work in the area of network analysis is discussed. Section III presents the network security analysis method using Big Data technology. Section IV presents the results and the final section concludes the paper.

II. Related work

Kaushik, Pilli and Joshi [17] proposed a method for network forensics which collects the network packets and extracts the features. The proposed method identifies and marks malicious packets based on the correlation between various network attacks and corresponding network parameters affected. A database of packets was created, and queries were created to identify and report attacks. However, the limitation of this method is the time taken to analyze and report the attacks when there are large or huge amount of data packets. The proposed method in this

978-1-5090-2246-5/16/$31.00 ©2016 IEEE

paper uses the Hadoop and Big Data technologies to perform queries on the large database of network packets. This greatly reduces the time needed to identify the types of attacks and report the attacks. This would make it possible to respond to the attacks in real time as soon as the attacks or anomalous connections are identified.

Siddiqui and Naahid [18] have worked on analysis of Knowledge Database and Data Mining Cup 1999 dataset employing k-means clustering algorithm using Oracle Data Mining technique to build the relation between protocols exploited by the attacker and attacks over the network. The pattern of attack types in relation with the various protocols exploited by the hacker is presented. Sun, Tao and Faloutsos [19] proposed a two-tier architecture, focusing on unsupervised learning that is capable of outlier detection. Clustering is used to group the anomalies based on similarities between them. The proposed approach in the paper is different in that it does not employ clustering or unsupervised learning method.

III. Network Security Analysis Using Big Data Technology

A. *The Proposed Network Security Analysis Method*

We have proposed a method for network security analysis using big data technology which includes five phases as shown in Figure 1.

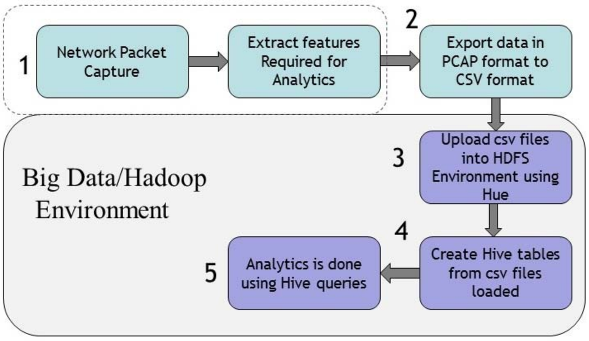
1
Network Packet Capture
Extract features Required for Analytics
2
Export data in PCAP format to CSV format
Big Data/Hadoop Environment
3
Upload csv files into HDFS Environment using Hue
4
Create Hive tables from csv files loaded
5
Analytics is done using Hive queries

Fig 1: Network Security Analysis Method

In phase 1 network traffic data is captured using Wireshark [7, 8, 9]. NCCDC dataset captured using Wireshark application is in PCAP format. Wireshark consists of an application programming interface for capturing network packets.

The following features have been extracted from the PCAP file: packet number, time stamp, source, destination, length, protocol, flag, and window size as shown in Table-1. The analysis of the data focused on detecting SYN Flood, SYN/FIN, SYN Scan, Null and Denial of Service attacks by employing Big Data and Hadoop technologies.

In phase 2, PCAP data with the features extracted using the customized fields in Wireshark application is exported to CVS format to enable exporting the data into Hive database application setup in HDFS environment.

In phase 3, data in CSV format is uploaded into the HDFS environment. Data can be uploaded/exported into HDFS environment using Command Line Interface or Hue service [15, 16]. In phase 4, the data uploaded into HDFS environment is imported into the table – log_data, created in Hive database. Log_data table schema is shown in Table 2. In phase 5, analysis is done using Hive queries to identify network attacks which is explained in the next section.

TABLE I. Features extracted from network traffic packets using Wireshark application

Features	**Description**
No	The number of the packet in the captured file
Time	The time stamp of the packet
Source	The IP address from where the packet is transmitted
Destination	The IP address to where the packet is transmitted to
Protocol	The type of protocol associated with the packet
Flag	Customized column: TCP.flag
Window Size	Transmission control protocol window size
Info	Additional information about the packet content

TABLE II. Log_data Hive Table schema

Column	**Data Type**
Packet No	int
Time	time
Source	String
Destination	String
Protocol	String
length	int
Flag	int
Info	String

B. *Attack Detection Based on Hive Queries*

Flag field in CSV file was in hexadecimal format and was converted into decimal format while dumping the data into Hive database. Queries were constructed to detect Denial of Service, SYN Flood, SYN Scan, Null scan attacks mainly carried out on TCP protocol. To construct the queries to detect the attacks the correlation between the TCP attacks and protocol features was used. Table III presents the correlation between the type of attacks and parameters in TCP [17].

TABLE III. Correlation between attacks in TCP protocol and parameters

Types of Attacks on TCP Protocol	Parameters
SYN Flood	SYN & ACK TCP Flag = 18
NULL Scan	TCP flag = 0
XMAS Scan	URG, FIN, PUSH TCP flag 41
SYN/FIN Attack	SYN & FIN TCP flag = 3

SYN Flood attack [10, 11] attacks transmission control protocol by establishing many half-open connections [12] and leaving no ports on the server to service the requests of new clients. The client machine requests a connection to the server by sending a SYN packet to the server. The server acknowledges the request for connection made by the client and in-turn sends a SYN-ACK packet to the client. At this stage, the server blocks a port to the corresponding client to establish a connection and waits for an acknowledgment from the client. To establish a successful connection the client must respond with an ACK packet to the SYN-ACK packet sent by server. The attacker explores this technique and implements SYN Flood attacks by not sending an ACK packet to the server. Sending numerous SYN packets to the server and not responding to the SYN-ACK requests sent by the server will lead to usage of all the resource leading to Denial of Service attack.

When a SYN-ACK packet is sent to the client from the server the TCP flag is set to '18' and if a client responds with ACK packet for a SYN-ACK packet received from the server the TCP flag is set to '16'. Hive query is constructed to explore the number of source IPs that made a SYN request from the client and received a SYN-ACK from the server and never responded to the server with ACK packet for more than 5 times are treated as malicious connections trying to implement SYN Flood attack. Below is the Hive query developed to detect SYN Flood attack.

Hive query to detect SYN Flood Attack

```
select      A.destination   AS   Source_IP,
A.source AS Destination_IP, count(no) from
log_data  A where flag = 18
and A.destination not in (select B.source
from log_data B where flag = 16 )
group by A.source, A.destination
having count(no) > 5
```

Xmas scan is adapted by the attackers to determine if the ports, on the server machine or any targeted machine on the network, are closed. Port scans on the targeted machine are done by sending TCP packets with all the flags set in the packet header to the targeted IP address. Transmitting packets with all flags set is illegal as per RFC 793 standard [21]. Any out-of-state packet with all TCP flags set and transmitted to open port at target machine is discarded. However, out-of-state packet with all TCP flags set and transmitted to closed port is responded with a RST response. This kind of behavior will allow the attacker to gain knowledge of closed ports on the targeted machine. Here, in the proposed network security analysis method, the packets in which all the flags are set and transmitted over the network are detected. When huge amount of traffic with all the TCP flags set being transmitted over the network is detected, it indicates Xmas scan. As a result, security application could be enabled to block such connections and secure the network. In TCP protocol when all the flags are set, the TCP.flag value is 41 in decimal format and hive query to detect Xmas scan attack is developed and shown below.

Hive query to detect XMAS Scan attack

```
select source, count(no) as Packets
from log_data where flag = 41
group by source
```

Null scan is similar to the Xmas Scan. In Xmas Scan all the flags in the packet header that are transmitted under TCP protocol are set and transmitted to the targeted machine. In Null Scan, the packets are transmitted with no flags set. When such packets are transmitted to the open ports of the targeted machine, they are discarded. However, when no-flag set TCP packets are transmitted to closed ports, they are responded with RST response from the targeted machine. This would enable the attacker to detect the closed ports on the targeted machine. Detecting and preventing huge amount of traffic of such nature over the network would prevent intrusion. When a packet with no flag set is transmitted, the code associated with TCP.flag is set to '0'. Below is the Hive query to detect Null Scan attacks over the network.

Hive query to detect Null Scan Attack

```
select source, count(no) as Packets from
log_data where flag = 0
group by source
```

IV. Experiment & Results

The size of the sample dataset (NCCDC dataset) is 131MB and has more than 1.2 million packets. All the packets are exported to Hadoop Distributed File System. Hive database was created to import the packets and analyze the data by constructing Hive queries. Queries were developed to identify the SYN Flood, Xmas Scan, and Null Scan attacks. Fig. 2 shows the amount of network packets of different network protocols in NCCDC sample dataset.

Queries were executed on the dataset imported into the Hive database to detect malicious packets in TCP traffic which would result in SYN Flood, Xmas scan and NULL scan attacks.

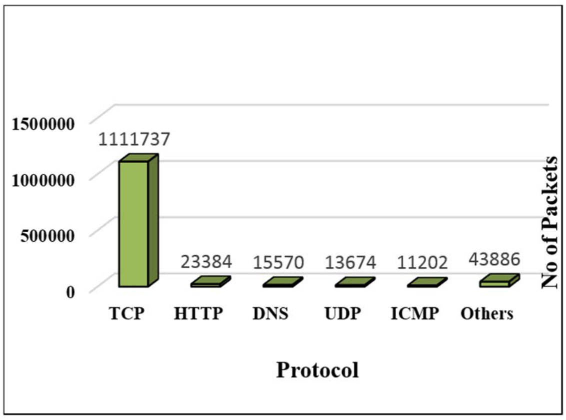

Fig. 2. Classification of Network packet from NCCDC sample dataset among different protocols

In the sample dataset, with 1219454 packets, there are 998 malicious packets resulted when SYN Flood detection query was executed on the database. 76 and 61 malicious packets resulted when Null Scan detection and XMAS scan detection queries were executed. Results are represented in Fig. 3. The queries would also extract the source IP address which has transmitted the malicious packet and the number of packets transmitted. This would help in preventing the specific IPs that transmitted the malicious packets from getting connected to the network.

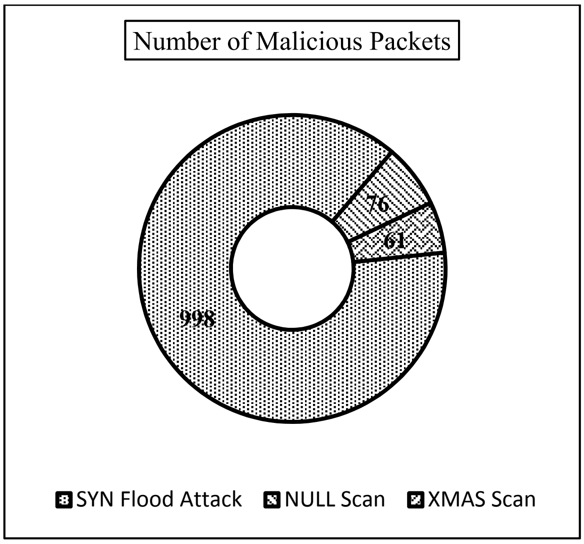

Fig. 3. Number of malicious packets detected in NCCDC sample dataset of 1219454 packets

Detecting the transmission of malicious packets and the type of attack that is being carried out in real-time would enable the prevention of further attacks being carried out. To test the performance of the proposed approach, Hive queries are executed against three sample datasets with different sizes and the time taken to detect the malicious packets was calculated. The first dataset is of 48 MB consisting of 404636 packets, the second dataset is of 131 MB consisting of 1219454 packets, and the third dataset consists of 3195010 packets. Fig. 4 shows the time taken to detect malicious packets on the three datasets with different sizes.

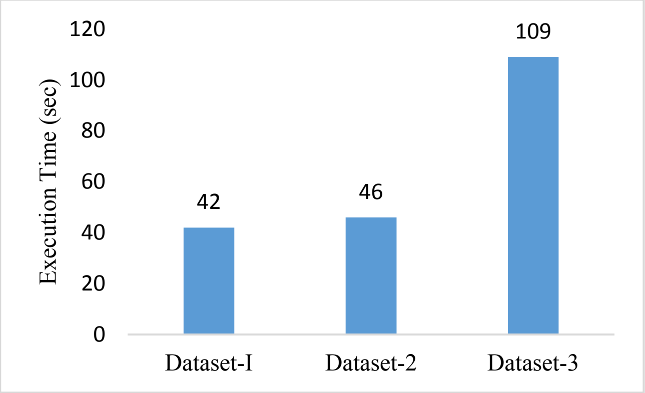

Fig. 4. Time taken to detect malicious packets

V. Conclusion

In this paper a new approach to perform network security analysis using Big Data technology is introduced. Features were extracted from network traffic capture (PCAP file) and exported to csv file using Wireshark. The csv file is then uploaded to HDFS environment and into Hive database. Analysis is then conducted using Hive queries. This approach was applied on sample datasets that were part of NCCDC dataset, and SYN Flood, Xmas scan, and Null scan attacks were detected. Using such an approach has the advantage of being able to analyze large amount of data and in short amount of time. This will enable real-time detection and prevention of network attacks.

Acknowledgment

This work is partially supported by NSF under the grant CNS-1460864. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF.

References

[1] W. Stallings and W. Stallings. *Cryptography and Network Security, 4/E*. Pearson Education India, 2006.

[2] J. Dean and S. Ghemawat. "MapReduce: simplified data processing on large clusters." *Communications of the ACM* 51.1 (2008): 107-113.

[3] J. Shafer, S. Rixner and A. L. Cox. "The hadoop distributed filesystem: Balancing portability and performance." *Performance Analysis of Systems & Software (ISPASS), 2010 IEEE International Symposium on*. IEEE, 2010.

[4] K. Shvachko, *et al.* "The hadoop distributed file system." *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*. IEEE, 2010.

[5] R. Lämmel. *"Google's MapReduce programming model—Revisited."Science of computer programming 70.1 (2008): 1-30.*

[6] J. Dean and S. Ghemawat. "MapReduce: a flexible data processing tool." *Communications of the ACM* 53.1 (2010): 72-77.

[7] G. Combs. "Wireshark." Web page*: http://www. wireshark. org/last modified* (2007): 12-02.

[8] C. Sanders. *Practical packet analysis: Using wireshark to solve real-world network problems*. No Starch Press, 2011.

[9] U. Lamping and E. Warnicke. "Wireshark User's Guide." *Interface* 4 (2004): 6.

[10] W. M. Eddy, "SYN Flood Attack." *Encyclopedia of Cryptography and Security*. Springer US, 2011. 1273-1274.

[11] T. Darmohray and R. Oliver. "Hot spares for DoS attacks." *login: The Magazine of Usenix and SAGE* (2000): 2000-7.

[12] S. J. Templeton and K. Levitt. "A requires/provides model for computer attacks." *Proceedings of the 2000 workshop on New security paradigms*. ACM, 2001.

[13] A. Thusoo, et al. "Hive: a warehousing solution over a map-reduce framework." *Proceedings of the VLDB Endowment* 2.2 (2009): 1626-1629.

[14] A. Thusoo, et al. "Hive-a petabyte scale data warehouse using hadoop." *Data Engineering (ICDE), 2010 IEEE 26th International Conference on*. IEEE, 2010.

[15] Cloudera.com,. "The Hue Service". N.p., 2015. Web. 24 Dec. 2015.

[16] Hue – Hadoop User Experience – The Apache Hadoop UI,. (2014). *How to configure Hue for your Hadoop cluster – Hue - Hadoop User*. Retrieved 24 December 2015, from http://gethue.com/how-to-configure-hue-in-your-hadoop-cluster/

[17] A. K. Kaushik, E. S. Pilli, and R. C. Joshi. "Network Forensic Analysis by Correlation of Attacks with Network Attributes." *Information and Communication Technologies*. Springer Berlin Heidelberg, 2010.

[18] M. K. Siddiqui and S. Naahid. "Analysis of KDD CUP 99 dataset using Clustering based Data Mining." *International Journal of Database Theory and Application* 6.5 (2013): 23-34.

[19] J. Sun, D. Tao, and C. Faloutsos, "Beyond streams and graphs: dynamic tensor analysis," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 374-383.

[20] PREDICT: Protected Repository for the Defense of Infrastructure against Cyber Threats. Available at :

https://www.predict.org/Default.aspx?tabid=104.

[21] J. Postel. "Transmission control protocol." (1981). Available at: https://tools.ietf.org/html/rfc793