Big Data Analytics Framework for Peer-to-Peer Botnet Detection using Genetic Algorithms

Mr. Prasad Koti,
Assistant Professor
Department of Computer Science,
Sarada Gangadharan College,
Velrampet Puducherry

Mrs. J. Madhupriya
Head of PG
Department of Computer Science,
Sarada Gangadharan College,
Velrampet Puducherry

***Abstract*— In recent days, Botnet attacks are the major problem faced by the security investigators and scientists on worldwide. Free software like Hadoop, Hive and Manhout are used to implement a quasi-real-time intrusion detection system The main desire is to identify botnet attack in Peer-to -Peer network by taking the advantage of expert systems. The paper is presented as follows: (1) Evolving a distributed scheme for the purpose of sniffing and Processing network traces to bring out the features of dynamic network with the help of Hive; (2) A decision model found on Genetic Algorithm is built to deal the complication of Peer-to-Peer Botnet detection in quasi-real-time.**

Keywords*— Botnet detection, Peer to Peer Networks, Machine Learning Algorithm, Genetic Algorithm

I. INTRODUCTION

Botnets constitutes a fatal cyber security deterioration that planned to hinder their harmful unsafe actions inside legitimate Internet traffic. The reason behind the popularity of botnet is the feature of regularly changing themselves over time, as a consequence of reacting to enhanced detection mechanisms. Futher more, Internet common communication protocols (i.e. HTTP) are used for resolving the underground communication medium. The unusual behaviours in varied botnets is distinguished by taking the advantage of Machine Learning algorithms (Genetic programming) That is to say, botnets duplicate the authentic HTTP traffic while exactly serving botnet purposes. At this edge, Zeus, Concker and Torpig botnets are distinguished by the evaluation based on the two specific feature sets. Definite propositions may be made concerning the beneficial of diverse feature sets and machine learning algorithms for locating all botnet without any exceptions.

Botnets signifies a group of compromised hosts directly below the management of a botmaster remotely i.e., a master slave interrelation [1]. They symbolize a technique of setting what would ordinarily be taken into account from authentic consumers to mischievous ends. Botnet behaviors are determined by the detection approaches as recommended by the investigators, the bots are upgraded by the botmasters to make impossible to detect. As the shared qualities of botnets are known, there are large number of approaches through detection and evasion may take place. Besides, given the traditional systems are boundlessly used, which dealt with linked to the Internet, old botnets also survive powerful. Consequently even after subsequent assessments, botnets remarkably revival[2]. Substantially, the lack of suitable advances in legacy systems results the restoration of botnets recovery of a botnet is potentially caused by the lack of necessary enhancements to traditional systems. As proclaimed, 104 devices was identified at the James A. Haley Veteran's Hospital near Tampa in the year of 2013. Likewise, McAfee foreseen that in 2014 attackers desire to aim systems with the support of old Windows XP operating system; where Windows XP is generally used in all traditional point of marketing and medicinal systems.

Large number of ongoing way are available to reveal botnet detection build on network traffic behaviour analysis. Considerable number of approaches make use of Machine Learning (ML) techniques (i.e. classification and clustering) which itself produce botnet revelation schemes. In such setup, the early move is to symbolize the network traffic in a manner that is worthwhile for the ML approaches used. To this edge, diverse systems consider the set of features on their own. Fewer uses network packet headers, at the same time remaining others demands packet payloads. Botnets, uses encryption techniques to avoid detection systems that investigate the communication information encapsulated in the packet payload. Fast-flux service networks, used mainly in centralized botnets, are a increasing trend in botnets. The use of fast-flux services makes botnet detection and mitigation harder. Apart from fast-flux, many botnet developers employs encryption to prevent intruders from reading the contents of C&C communication.

The main objective is to make an attempt to surpass C&C traffic of a P2P botnet. Forming a P2P network is a challenging task that comes with many problems. Programmers commonly choose to implement either a structured or an unstructured overlay network. Structured networks have the drawback of higher overhead. On the other side, unstructured networks, because of their routing mechanisms, such as flooding or random walk, can leads to massive query replication [26]. This is not desired in botnets, because it can lead to detection. The effective performance of a P2P network is based on a better and reliable query routing algorithm, network adaptation to unresponsive peers and bootstrap procedure algorithm. For this simulation, a

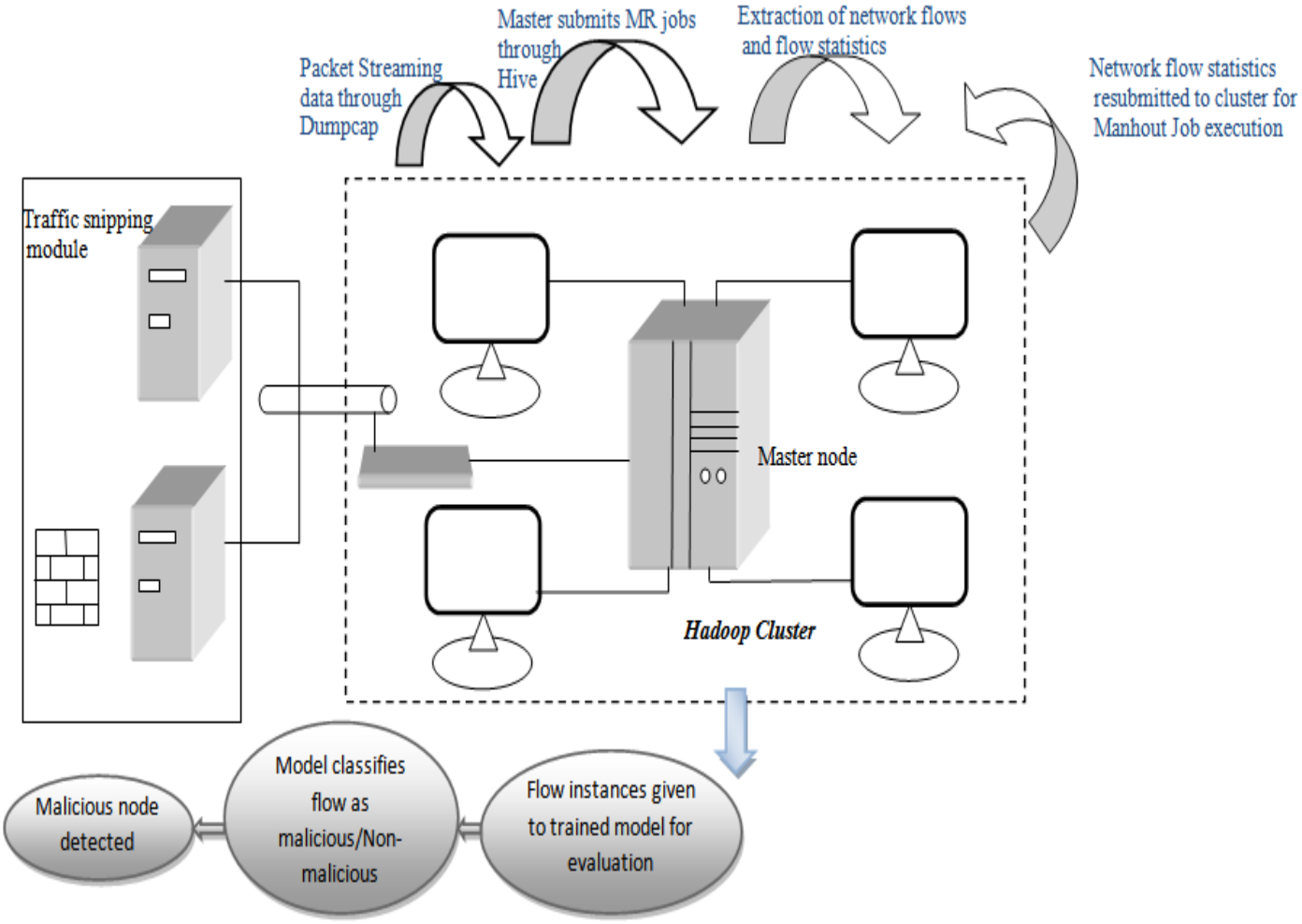

Fig 1. Overview of scalable P2P botnet detection framework

structured P2P network will be more appropriate than an unstructured network. This is because of the requirement of the guarantee of rapid query resolution even for rare items. In structured systems, queries are normally resolved in sublinear time (O(logN)). We also want to avoid wide range of query traffic, which is typical for unstructured networks. We decided to use our adapted version of the Kademlia protocol, as it is relatively easy to implement and several successful botnets are known to use this protocol, for example the Storm botnet.

An important obstruction of P2P network implementation is a dedicated connection between two peers separated by NAT or firewall. There is a solution, intended mainly for UDP protocol. However, this solution needs a server with a public IP address that acts as a rendezvous point for the peers. As this would be very to execute, our simulation will be modeled to run either on a single local network or on hosts with global IP addresses.

On the same way as the previous simulation of an HTTP botnet, this simulation should perform observations of C&C traffic, mainly for development of new security algorithms. However, it can also serve as a simulation of a structured P2P network leveraging the Kademlia protocol.

The main goal of this work is to provide an understanding into the current state of malware communication. As security solutions are gradually developing, malware developers requires the development of more sophisticated communication models to avoid detection. Malware has various forms: trojan horses, worms, viruses, spyware or botnets, to name a few. This work mainly focuses on botnets in particular, because they represent one of the most dangerous forms of malware and more frequently use sophisticated communication channels.

II. STRUCTURE OF P2P BOTNET DETECTION

The outline of extensible P2P botnet detection schema is presented in Fig 1 [3]. The three components in the framework are:

1. Traffic Sniffer Module for preliminary processing of packets

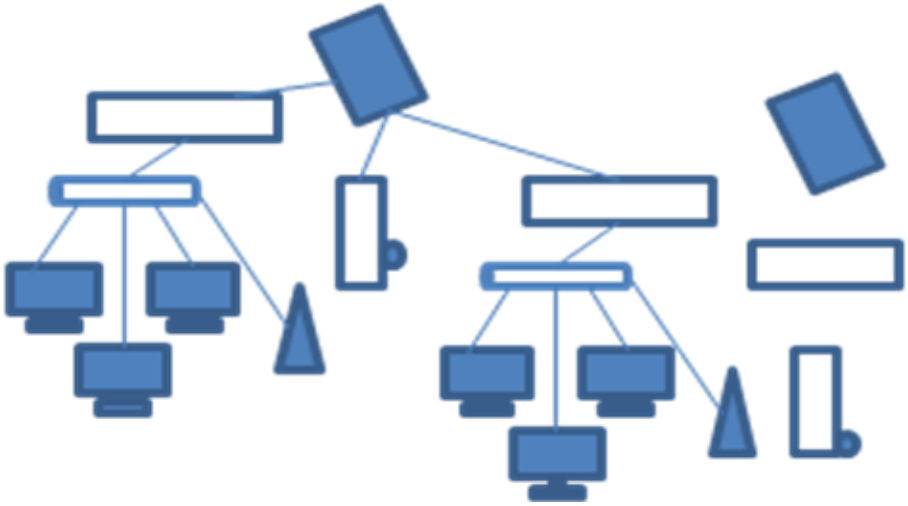

Fig 2. Test bed for obtaining mischievous network action

2. Feature Extraction Module for building feature set, and

3. Machine Learning Module for studying and investing the mischievous traffic

A. Traffic Sniffer Module

Dumpcap [4] sniffs the packets from the network interface meanwhile Tshark [5] provides the service of obtaining the 172 fields, establish a link to feature set, out of the packets and give the fields to the Hadoop Distributed File System (HDFS). It holds the traffic from the wire into successive pcap files of a particular size through capture ring buffer option. At the same time, few of the packets are discarded and are not recorded to the file system. some analysis were made by differing the size of the buffer. Feature Extraction Perl Scripts are used to computerize the unified Traffic sniffing which run as daemon jobs and originate numerous occurrences of Tshark at regular time intervals, each deriving the field mandatory for feature set formation from the concluded pcap file.

An unique and private set of Linux systems are used as test bed for collecting the mischievous sources for the experimentation. the private network is organized by linking the systems to an access switch. On top of every physical machines, the authors ran virtual machines with Windows XP as the operating system. This setup is shown in Fig. 2.

B. Feature Extraction Module

Apache Hive performs screening of features from the delimited files given to HDFS. Once the delimited files are submitted to HDFS. Apache Hive and Tshark enables to vary the feature at runtime is considerable feature. The Feature Extraction Perl Script, mentioned above, grants the user to characterize the fields to extract from packet with the help of Tshark and then generates the table in Hive correspondingly. This provides adaptability to extract distinct features applicable to discrete problem instances and also avert the diligent task of altering the whole code, in case the features were obtained via MapReduce program, which was seen in previous study. In case a different feature set is chosen in this script, a different table is undoubtedly generated.

C. Machine Learning Module

For achieving scalability in Machine Learning Module, a machine learning library which is constructed on the top of Hadoop called Manhout is used. Considering every individual core algorithms for categorization and gathering are run as MapReduce jobs, the high computational power of the cluster is controlled to obtain improved outcomes.

GA s are simple but provide most efficient results for the problem domain on natural selection. These search algorithms performs a step-by-step paradigm to identify a pool with better fitness. Moreover GA rely on Genetic Models and since computationally less complex. GA are gifted with many features. These features allows to form various classes of approaches for Genetic method.

Simple GA involve only copy operation and exchanging certain features of parents. It promises more optimized results with less computational complexity. For categorization, rule based models are widely constructed based on GAs. Rather than physically writing the rule set, as required by classical rule-based system, the concept of using GAs for categorization is to automatically generate such classification rules.

III. CONCLUSION

The contribution of the paper are as follows:

1. An extensible packet capture module to measure data of large capacity in a quasi-real-time (within 5–30 s delay).

2. A shared dynamic feature extraction structure to define flow statistics of packet captures.

3. A Peer-to-Peer security threat detection module which segregates mischievous traffic on a cluster.

Further research could follow up by mapping the host-level behavior of botnets and their propagation in order to get a full perspective on their life. It would also be interesting to see the revenue coming from today's botnets, which was mentioned only briefly in this paper. This could shed light on the motives of botnet developers.

REFERENCES

[1] SCMagazine, WordPress2, 2013. <http://www.scmagazine.com/wordpress-attacks-showcase-botnet-owners-expanding-tricks/article/288947/>(accessed: 16.05.13).
[2] StackExchange,WordPressAttack,2013.http://security.stackexchange.com/questions/34482/understanding-what-bot-was-used-for-a-botnet attack> (accessed: 16.05.13).
[3] Kamaldeep Singh , Sharath Chandra Guntuku, Abhishek Thakur, Chittaranjan Hota ,"Big Data Analytics framework for Peer-to-Peer Botnet detection using Random Forests" Information Sciences, 2014
[4] Dumpcap, 2013. <http://www.wireshark.org/docs/man-pages/dumpcap.html>.
[5] TShark,2013.<http://www.wireshark.org/docs/man-pages/tshark.html>