iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 Issue: IX Month of publication: September 2014

DOI:

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Bot Detection using Traffic Monitoring and Traffic Analysis

R.Kannan[1], A.V.Ramani[2]

Associate professor[1], Associate Professor & Head[2]
Sri Ramakrishna Mission Vidyalaya College of Arts and Science
Coimbatore -20

Abstract: Botnet is peak extensive spread and occurs normally in today's cyber attacks, resulting in severe intimidations to our network possessions and group's properties. Botnets are congregation of negotiated computer (Bots) which are distantly precise by its creator(BotMaster) below a mutual Command-and-Control(C&C) infrastructure. They are used to dispense commands to the Bots for malicious actions such as distributed denial-of-service (DDoS) occurrences, junk and phishing. Maximum of the prevailing Botnet detection methods focus only on specific Botnet command and control (C&C) practices (e.g., IRC, HTTP) and structures (e.g., centralized), and can become vain as Botnets change their construction and C&C techniques.

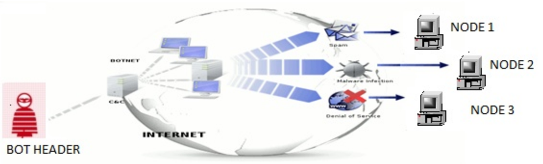
Keyword:IRC, C&C, DDoS, NSM, UBE, ARGUS

I. INTRODUCTION

In this paper, we proposed a two different Botnet detection method. The Botnet detection based on traffic monitoring and Botnet detection based on traffic analysis. The first method is a proposed frame work based on finding similar communication patterns and behaviors among the groups of hosts that are performing at least on malicious activity. The point that distinguishes our proposed detection frame work from many other similar works is that there is no need for prior knowledge of Botnets such as Botnet signature.In the second method we focused on bots using IRC to communicate and examined the behavior of such bots when they connected to an IRC server. We observed the actual traffic of some ports which were often used by IRC protocol. As a result, we confirmed that bots tried to reconnect to an IRC server at certain intervals when the server refused the connection from the bots. Moreover, we examined the distribution of the intervals and confirmed that the communication from other IP addresses showed similar behavior.

IRC(Internet Relay Chat)

When an attacker passes the instructions on to bots, IRC is generally used as a way of communications. IRC is a talking system that exchanges the client's messages for text data on the TCP/IP protocol through servers. It is possible to send instructions to many bots by utilizing the multicast delivery mechanism of IRC. It is a reason why attackers use IRC to transmit instruction to bots.When computers are infected with bots, they try to connect to an IRC server. The computer that connects with IRC participates in a specified channel, and waits for instruction. Instructions are communicated by attacker's messages. Bots execute instructions after bots are received them. The IRC server might be infected with bots and might be controlled. We cannot stop an entire Botnet if we can stop one IRC server. Therefore, it is difficult to the attacker.



We proposed a new general frame work for detection of Botnets that currently targets P2P based Botnets. However the frame work has the capability of adding another component for centralized Botnets. This framework at one stage monitors that group of hosts that perform at least one malicious activity and then try to find the hosts that show similar communication and behavior patterns. In the second method we aim to discover new features by monitoring and analyzing operations of bots. Bots need to connect with a server that passed on instructions when instructions are received from the attacker. Then, we monitored operations when clients connected it with a server. In the result, we observed common operations to doubtful clients. In addition, we investigated intervals of the time that the client communicated with the server, and observed similar patterns.

Detection based on traffic monitoring:

However, there are two essential techniques for Botnet Detection in traffic monitoring setting up honey nets and passive network traffic monitoring. Many papers discussed about using honey nets for Botnet detection. But we have to take into consideration that honey nets cannot detect Bot infection most of the times and are just good for understanding Botnet characteristics. For identifying the existence of Botnets in the network, passive network traffic monitoring is helpful. This technique can be classified into signature-based, anomaly-based, BNS-based, and mining based. Signature-based detection techniques can just be used for detection of recognized Botnets. Therefore, this solution is not functional for unknown bots. Anomaly-based detection techniques attempt to detect Botnets based on several network traffic anomalies such as high network latency, high volumes of traffic, traffic on unusual ports, and unusual system behavior that could indicate presence of malicious bots in the network. DNS-based detection techniques are based on DNS information generated by a Botnet. As mentioned before, bots normally begin connection with C&C server to get commands. In order to access the C&C server bots carry out DNS queries to locate the particular C&C server to get commands. In order to access the C&C server bots carry out DNS queries to locate the particular C&C server that is typically hosted by a DDNS (Dynamic DNS) provider. Therefore, it is feasible to detect Botnet DNS traffic by DNS monitoring and detect DNS traffic anomalies. Data mining techniques are also can be used to detect Botnets. Geobl and Holz proposed Rishi in 2007. Rishi is based on traffic monitoring for IRC servers, suspicious IRC nick names and uncommon server ports. They use ngram analysis and a scoring system to detect bots that use unusual communication channels, which are commonly not detected by standard intrusion detection system. This technique cannot detect non-IRC Botnets as well as encrypted communication. Masud etalproposed efficient flow-based Bots net traffic is applicable to non IRC Botnets. Because this method does not require access to pay load content, it is applicable even if C&C pay load is not available or encrypted.

Our proposed frame work is based on passively monitoring network traffics, architecture of our proposed Botnet detection system, which consist of 4 main components: Filtering, Application Classifier, Traffic Monitoring, Malicious Activity Detector. Filtering is responsible to filter out irrelevant traffic flows.

II METHODOLOGY

A. Filtering

Filtering is responsible to filter out irrelevant traffic flows. The main objective of this part is for reducing the traffic work load and makes the rest of the system perform more efficiently. Figure A shows the architecture of our filtering system in C1, we filter out those traffics which targets (destination IP address) are recognized servers and will unlikely host Botnet C&C server. In C2, we filter out traffics that are established form external host towards internal hosts. In C3, we filter out handshaking processes (connection establishments) that are not completely established. Handshaking is an automated process of negotiation that dynamically sets parameters of a communications channel established between two entities before normal communication over the channel begins. It follows the physical establishment of the channel and precedes normal information transfer.

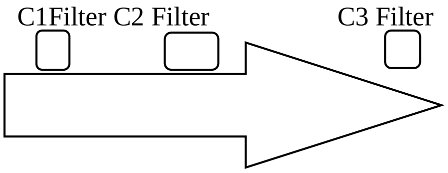


FIG. A

Traffic filtering stages

B. Application Classifier

Application Classifier is responsible to separate IRC and HTTP traffics from the rest of traffics and send them Centralized part. For detecting IRC Traffics we can inspect the content so each packet and try to match the data against a set of user defined strings. For this purpose we use pay load. Inspection that only inspects the first few bytes of the payload and looking for specific strings. These IRC specific strings are NICK for theClient's snick name, PASS for a password, USER for the user name, JOIN for joining a channel, OPER that says a regular user wants to become a channel operator and PRIVMSG that says the message is a private message. Using this strategy for detecting IRC traffic is almost simple for most network intrusion detection software like snort.

C. malicious Activity Detector

In this part we have to analyses the out bound traffic from the network and try to detect the hosts are performing at least one malicious activity. Each host may perform different kind of malicious activity but Scanning, Spamming, Binary downloading and exploit attempts are the most common and efficient malicious activities a bots master may command their bots to perform. In this paper we just focus on scanning and spam-related activities. The output show this part is the list of hosts which performed malicious activities.

Scanning: Scanning activities may be used for malware propagation and DOS attacks. There has been little work on the problem of detecting scan activities. Most scan detection has been based on detecting N events with in a time interval of T seconds. This approach has the problem that once the window size is known, the attacker scan easily evades detection by increasing their scanning interval. Sort are also use this approaches. Snort version2.0.2 uses tow preprocessors. The first is packet-oriented, focusing on detecting mal formed packets used for ”stealth scanning” by tools such than X number of ports or Y number of IP addresses with in Z seconds. Snort's parameters are tunable, but it suffers from the same drawbacks as Network Security Monitor (NSM) since both rely on the same solution for using in this part is Statistical Can Anomaly Detection Engine (SCADE). A snort processor plug-in system which has two modules, one for inbound scan detection and another one for detecting out bound attack propagation.

Spam-related activities: E-mail spam, known as Unsolicited bulk Email(UBE), junk mail, is the practice of sending unwanted email messages, in large quantities to an indiscriminate set Botnets. A number of famous Botnets which have been used specially for sending spam is Storm w or which is P2P Botnet. Our target here is not recognizing which mail message is spam, though for detecting group of bots that sending spam with detecting similarities among their actionsand behaviors. Therefore the content of emails from internal network to external network is not important in our solution. All we want to do is determining which clients have been infected by bot and are sending spam. For reaching to this target, we are focusing on the number of emails sending by clients to different mail servers. Based on our experience in our lab, using different external mail servers for many times by same clients an indication of possible malicious activities. It means that it is unusual that a client in our network send many emails to the same mail server (SMTP server) in the period of time like one day.

D. Traffic Monitoring:

Traffic Monitoring is responsible to detect the group of hosts that have similar behavior and communication pattern by inspecting network traffics. Therefore we are capturing network flows and record some special information on each flow. We are using Audit Record Generation and Utilization System (ARGUS) which is an open source tool for monitoring flows and record information that we need in this part., Each flow record has following information: Source IP(SIP) address, Destination IP(DIP) address, Source Port(SPORT), Destination Port(DPORT), Duration, Protocol, Number of packets(np) and Number of bytes(nb) transferred in both directions. Then we insert this information on a data base like then, we insert this two new values (nbps and nbpp) including SIP and DIP of the flows that have been marked into another database, similar to figure 4. Therefore during the specified period of time (6 hours), we might have a set of database, {di} i=1….m which each of these databases have same SIP, DIP, DPORT and protocol(TCP, UDP). We are focusing just at TCP and UDP protocols in this part.

As we mentioned earlier, the bots belonging to the same Bots net have same characteristics. They have similar behavior and communication pattern, especially when they want to update their commands from Bot masters or aim to attack a target; their similar behaviors and more obvious. Therefore next step is looking for group of databases that are similar to each other.

For finding similar communication flows among databases {di} = 1, one solution is using clustering algorithm like X-means clustering algorithm. We proposed a simple solution for finding similarities among group of databases. For each database we can draw a graph in x-y axis, which x-axis is the Average Number of Bytes per packet (nbpp) and y-axis is Average Number of Byte per Second (nbps). (X,Y) = (nbpp, nbps)

For example, in database (d i), for each row we have nbpp that specify x-coordinate and on the x-y axis graph. We do this procedure for all rows (network flows) of each database. At the end for each database we have number of points in the graph that by connecting those points to each other we have a curvy graph.

Next step is comparing different x-y axis graphs, and during that period of time (each 6 hours) those y axis graphs that are similar to each other. Each of these graphs is referring to their corresponding database in previous step. We have to take record of SIP addresses of those hosts and report them as possible bots in the network.

Detection based on traffic analysis:

We aimed to observe new features of IRC-based bots in this research. Firstly, we monitored the port which generally used by IRC. As the result, we observed that clients with specific IP address are different from other clients in terms of flow of connection to IRC server.

Data over view:

In the investigation, we used the traffic of port from 6666 to 6669. The traffic used in this paper is flowed in 24 hours. Action and features while connection to server:When an IRC client uses an IRC server, it is necessary to connect it with the IRC server. The produce of connection is provided. An IRC client transmits the command of NICK and USER to the server. The IRC server registers the client after receiving both commands. The connection of the client and the server establishes by being processed these commands. Secondly the client transmits the JOIN command to participate in the channel. The client can exchange messages mutually by participating in the channel. The PRIVMSG command or the NOTICE commands are used for it.Generally communication like this is following flows as

- NICK→USER→JOIN→PRIVMSG (or NOTICE)→..

However, clients with specific IP addresses are refused to connect to the IRC server. IRC servers refuse the connection by the clients with suspicious behaviors in order to prevent the nick name duplication, the overload to the server and the connection by the doubtful clients. Such clients repeat transmitting NICK and USER until the connection succeeds. Thus, the flow of the clients as follows.

- NICK→(ERROR)→NICK→(ERROR) →…
- NICK→USER→(ERROR)→NICK→USER→(ERROR) →…

III. EXPERIMENTAL RESULTS

The traffic on ports from 6666 to 6669 is used in this section, because these ports are usually used for IRC communication. We limited all traffic to traffic utilized IRC by seeing information on the protocol. In the interval of time that the client communicated with the server. The ratio of interval of communication is shown in figure 1-5. From figure 1 to figure 4 are the communications of doubtful clients X-axis is a logarithm display as for the number of seconds of communication interval. Y-axis is a ratio of each number of seconds to the whole. For example, we can see the ranges of 0.2-0.3 seconds account for about 50 percent’s of the whole in figure 1. In addition, we can see the communication of the ranges of 0.2-0.3 seconds account for 60 percent of whole in fig 2.in figure 3, the features are seen especially remarkably. In figure 4, distribution of communication interval concentrates about 30 seconds.

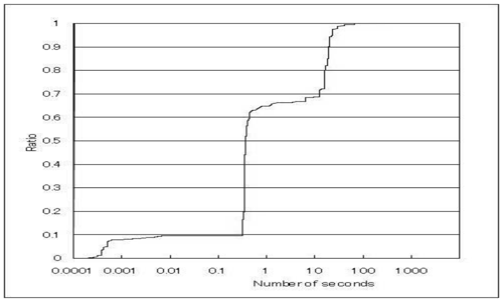


FIGURE1.RATIO OF COMMUNICATION INTERVAL TO IRC SERVER FROM CLIENT GUESSED TO THE BOT(PORT6667)

- Lastly, bias is in the three places of about 0.3 seconds, about 8 seconds, and 400 seconds. In detail, figure1 is a communication used port 6667.figure 2, 3 and 4 are port 6666, 6668 and 6669 respectively.

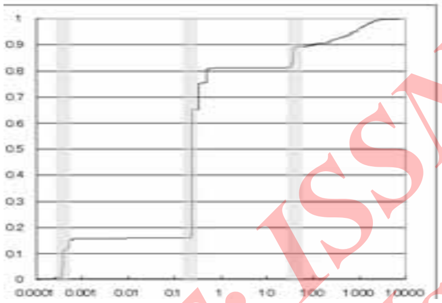


FIGURE2.RATIO OF COMMUNICATION INTERVAL TO IRC SERVER FROM CLIENT GUESSED TO THE BOT(PORT6666)

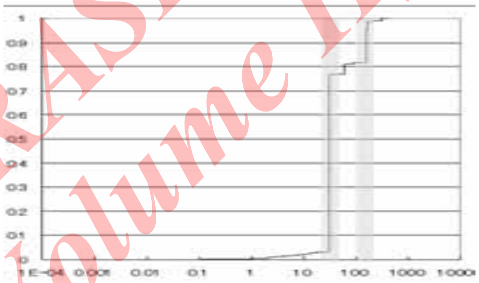


FIG 3.RATIO of communication interval to *IRC SERVER FROM CLIENT GUESSED TO THE BOT (PORT6668)*

We consider that there are various advantages by making traffic visible like these.

- Judgments of kind of bots more than past viruses
- Automation of detection of bots by machine studies
- Future works are as follows.
- Investigations of more objects(problem of verification and generality)
- Considerations of detection by concrete visible
- Measures against bots that does not use IRC

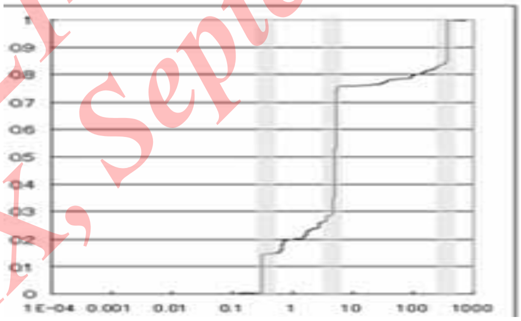


FIGURE4.RATIO OF COMMUNICATION INTERVAL TOIRC SERVER FROM CLIENT GUESSED TO THE BOT (PORT6669

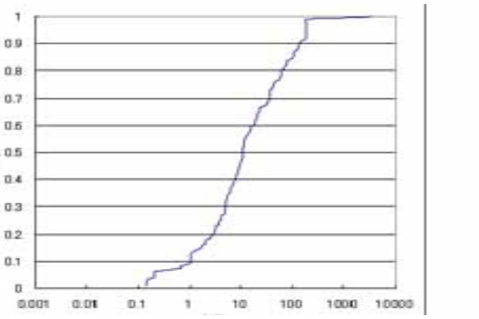


FIG 5 RATIO OF COMMUNICATION INTERVAL AT IRC SERVER FROM CLIENT GUESSED NOT TO BE BOT

Figure 5 is shown ratios of communication interval. We can see the bias of distributions is few compared with figure 1-4 which are shown the distributions of it to a client guessed not to be Bot.

Consideration

In this chapter, we presumed that there are differences between IRC client by user and bots, and we examined traffic port used by an IRC server generally. We indicated that how much time the communication interval between a client and an IRC server. As a result, we observed common features of client's traffic guessed to be bot. We confirmed that there is a bias of intervals of communication to the server in any traffic. In addition, there are differences of the distributions bias and the number of time in common features.

IV. Conclusion

In comparison to other kind of malware Botnets are harder to monitor a shutdown and detection of them becomes challenging problem. In this paper we proposed a new general detection frame work and we aimed to examine new features on Bot detection methods on features of Bot behaviors. In our proposed detection frame work, we monitor the group of hosts that perform at least one malicious activity in one step and then try to find the hosts that show similar communication patterns among them. The point that distinguishes our proposed detection frame work from many other similar works is that there is no need for prior knowledge of Botnets such as Botnet signature.

Where as in analysis we consider the features of behaviors of bots based on IRC. Secondly, we captured the traffic of port used by IRC, and examine in what patterns the feature was seen. As a result, we observed that clients guessed to be Bot stake different communication patterns compared with clients guessed not to be bot. In adding, we plan to further improve the efficiency of our proposed detection frame work with adding unique detection method in centralized part and make it as one general system for detection of Botnet and try to implement it in near future.

References

[1] Botnet attacks and historical lists of Bot nets http://en. Wikipedia.org/wiki/Botnet

[2] Time to live on the network. Avantgarde Marketing&Design,http://www.avantgarde.com/xxxxttln.pdf.

[3] Honey net project, know your Enemy: tracking Botsnets.http://www.honeynet.org1papers /bots

[4] J. R Binkley and S. Singh. An algorithm for anomaly-based Botnet detection. In proceedings of USENIXSRUTI'06, pages 43-48, July 2006.

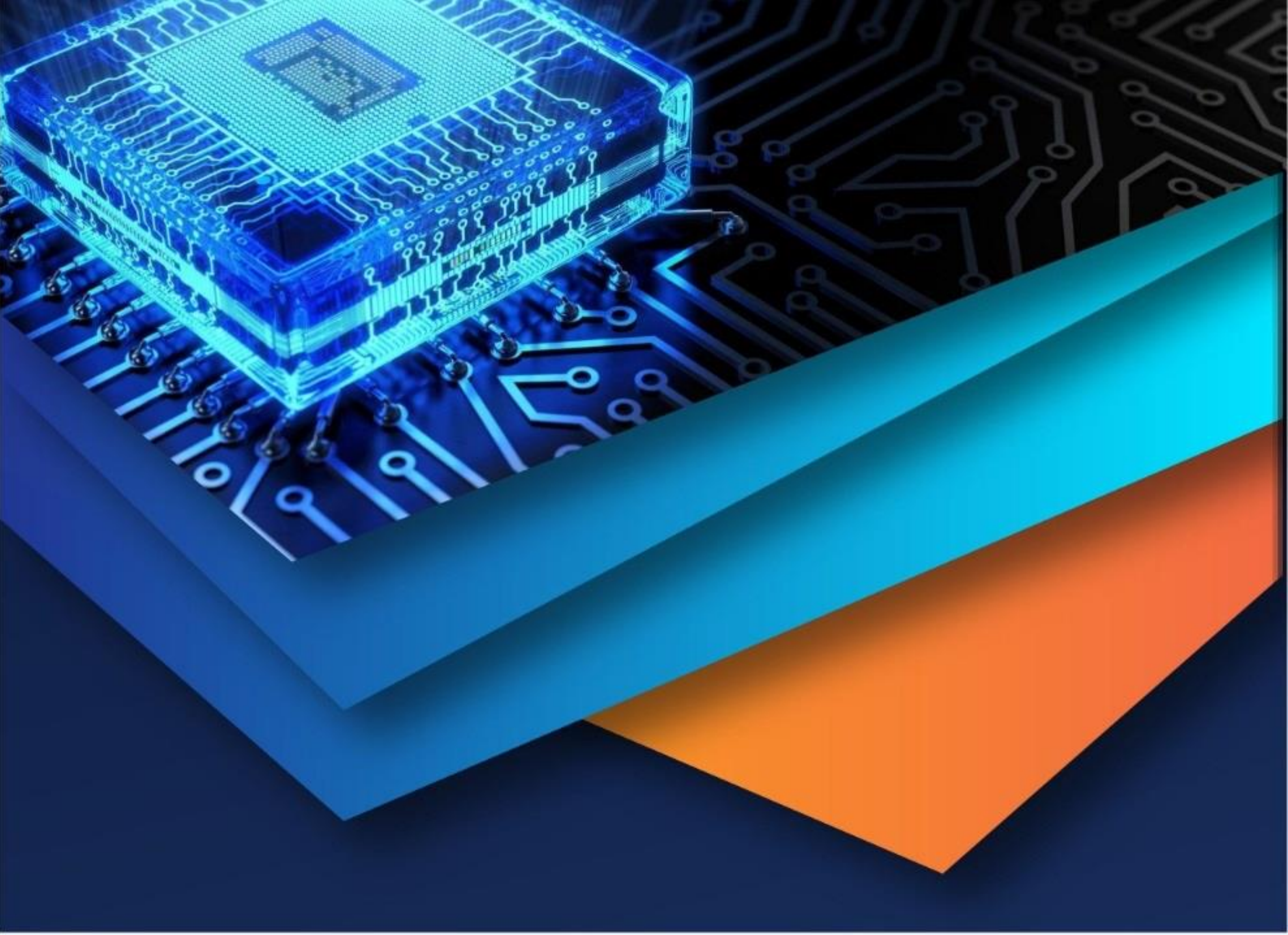
[5] Gu G, Zhang J, Lee W. "BotSniffer: detecting botnet command and control channels in network traffic". In: Proceedings of the 15th annual network and distributed systemsecurity symposium 2008

[6] W. Timothy Strayer, David Lapsely, Robert Walsh, Carl Livadas. "Botnet Detection Based on Network Behavior" In: Advances in Information Security, Springer, Volume 36, 2008, pp 1-24

[7] The Honeynet Project. French Chapter [Online] http://www.honeynet.org/chapters/france.

[8] David Zhao, IssaTraore, BassamSayed, Wei Lu, SherifSaad, Ali Ghorbani and Dan Garant."Botnet detection based on traffic behavior analysis and flow intervals", Computers and Security in year 2013.

[9] R.Kannan, Dr.A.V.Ramani, "Flow Based Analysis to Identify Botnet Infected Systems", Journal of Theoretical And Applied Information Technology (E-ISSN 1817-3195 / ISSN 1992-8645), Vol 67 September 2014.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429