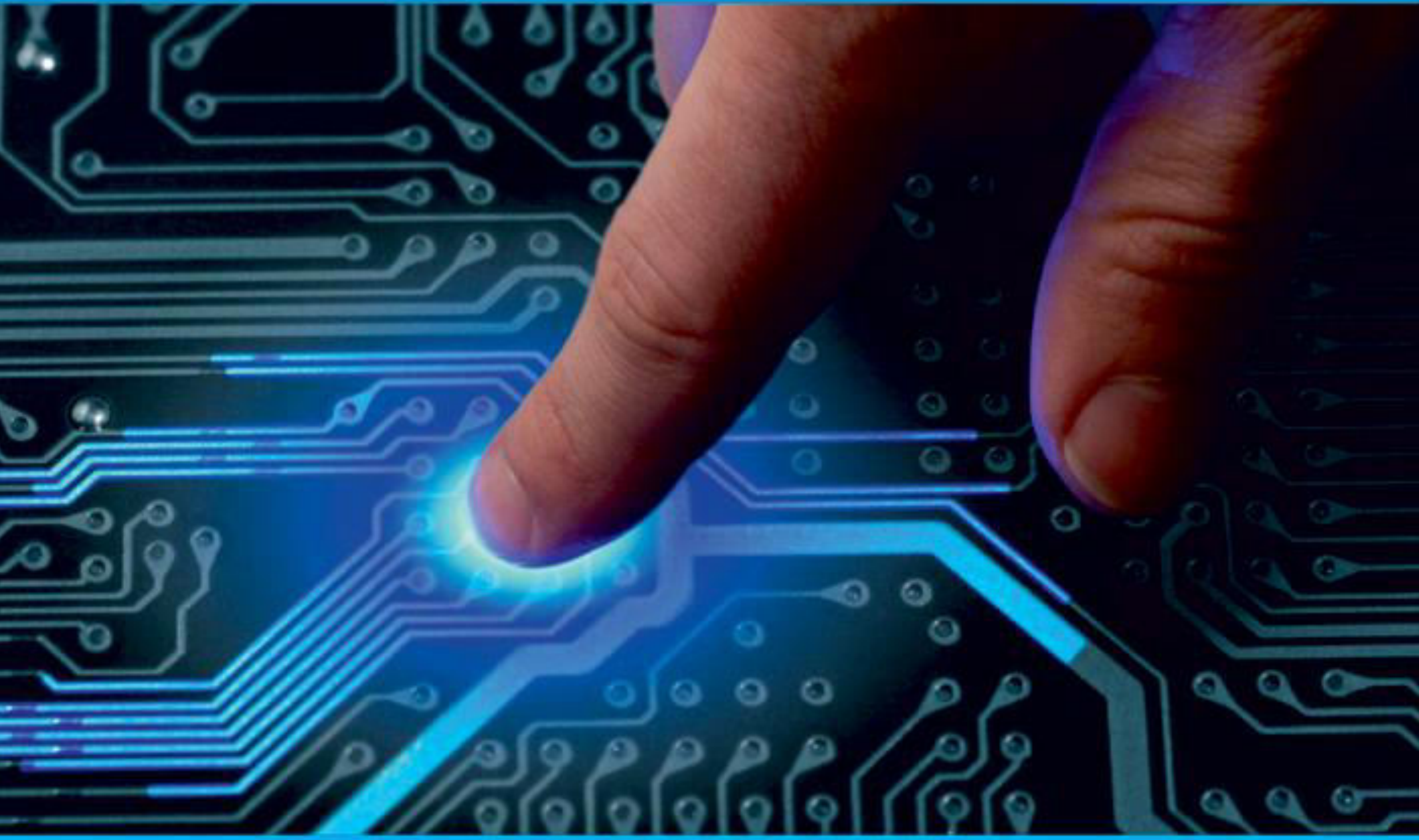
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 10, October 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.625

© 2024 IJIRCCE | Volume 12, Issue 10, October 2024| DOI: 10.15680/IJIRCCE.2024.1210058

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Bot Detection in Motion: Real-Time Network Traffic Insights

Dasharath Chaudhary[1], Maharshi Dave[2], Kiran Dodiya[3], Khunt Akash[4], Divya Patel [5]

M. Sc Cyber security, NSIT-IFSCS Jetalpur, Ahmadabad (Affiliated to National Forensic Sciences University) Gandhinagar, Gujarat, India[1]

MSS Engineer, Indusface, Vadodara, Gujarat, India[2]

Assistant Professor & Program Coordinator of DFIS (Cyber Security &Digital Forensics) NSIT-IFSCS (Affiliated to National Forensic Sciences University), Gandhinagar, Gujarat, India[3]

Assistant Professor & Program Coordinator of Cyber Security (Cyber Security &Digital Forensics) NSIT-IFSCS (Affiliated to National Forensic Sciences University), Gandhinagar, Gujarat, India[4]

Assistant Professor &amp; Course Coordinator of DFIS (Cyber Security Digital Forensics) NSIT-IFSCS (Affiliated to National Forensic Sciences University), Gandhinagar, Gujarat, India[5]

ABSTRACT: As the adoption of AI-driven technologies accelerates, digital systems become more efficient and vulnerable to sophisticated cybersecurity threats, particularly from malicious bots. These bots can operate in different environments, such as web browsers or system processes, executing harmful activities like distributed denial of service (DDoS) attacks, data scraping, and credential theft. This research introduces a comprehensive, data-driven bot detection and mitigation framework that operates in real-time, independent of the type or origin of the bot. The framework leverages machine learning algorithms and a multi-layered analysis of network traffic logs, enriched by several data sources, to identify and mitigate bot activity. Key detection methods include: (1) User-Agent String Analysis, where the system examines HTTP headers and user-agent strings to match patterns typically associated with browser-based bots; (2) Source IP Analysis, which tracks and analyses IP behavior over time to identify bot networks or repetitive suspicious access attempts; and (3) Sample Behaviour Analysis, which collects detailed behavioral data from detected bots to understand their operational patterns, movement within systems, and potential attack vectors. The framework's detection engine is reinforced by a Bot Log Database, which stores historical data on previously detected bot behaviors, user-agent strings, IP addresses, and attack patterns, providing a reference point for future detection. This data-centric approach allows the system to detect various bots, including browser, system-level, chat, transactional, and download bots. Once a bot is identified, automated mitigation techniques are applied, including generating dynamic firewall rules that block malicious traffic by IP addresses or ports. The use of real-time log capture, combined with machine learning and an extensive database of bot activity, enables a proactive, scalable, and efficient approach to bot detection and prevention. By continuously updating its database with new bot behavior data, the framework remains adaptive and capable of neutralizing evolving bot threats, ensuring the security and integrity of digital systems across diverse environments.

KEYWORDS: Bot, network traffic, user-agent, user string, machine learning, database, log data.

I. INTRODUCTION

1.1 Overview

In this era, everything is working on digitalization, and people are now using super technology or AI-integrated technology for fast and effective work.[1].

In the network, many services and technologies are integrated with AI that can increase data breaches or malicious user activities via DOS or DDOS attacks.[2]. In these attacks, attackers use scripted malicious Bots on search engines or high-traffic websites. Header Connect Bot for Honeypot users.[3], [4]. In the network, many types of suspicious or malicious things are running.[5]Malicious activity identification is critical in the network. It cannot be mitigated because it is not

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

easily detected in the network/system, and system protection measurement cannot catch those activities without using proper standard tools and techniques.[6]. Many bots run on web browsers or software-patched scripted bots that run Internet access. [7]Their system protection methodologies or services cannot catch that because it is not detectable without proper skill-based domain expertise knowledge or specific techniques.[8].

1.1.1 What is a Bot?

The bot is a software-based scripted program or software that works on the internet to perform a specifically defined task.[9]The bot runs on the Internet, is automated, and performs various tasks that can be good or bad.[10][11].

1.1.2 Types of Bots

There are many types of bots running on the network, including web crawlers (used for search engines), chatbots (conversation use), transaction Bots (perform various transitions), social media Bots (social content sharing), scarper Bots (collect data from websites), game Bots (online games perform a task), customer service Bots (customer service and problem-solving tasks), and many types of Bots running on the network.[9].

1.1.3 Classification of Bots

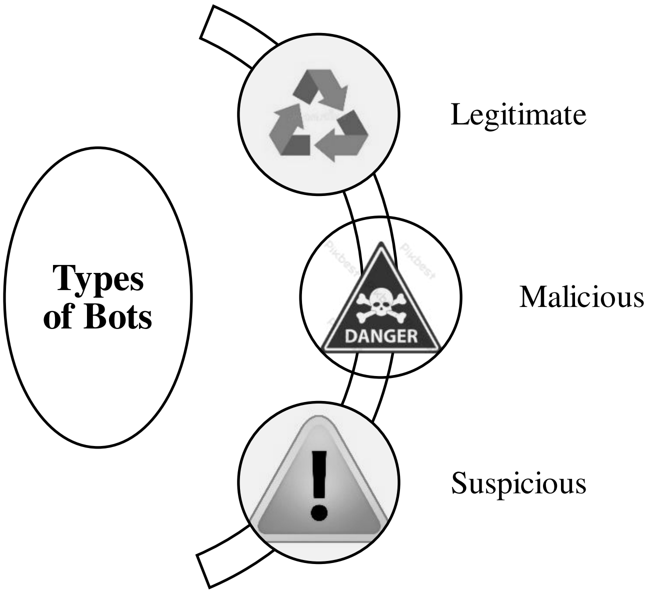


Figure 1 Classification of Bots

Bots are classified according to their use-based perspective into three types: legitimate (use legal task perform), malicious (use illegal activity task perform), and suspicious (not defined can be used for legal or unlawful perspective for task perform).[12], [13].

II. LITERATURE SURVEY

The chart systematically summarizes various research studies that contribute to botnet detection. It organizes these studies into key components, highlighting their methodologies, objectives, challenges addressed, and performance metrics.

2.1. Study

This section identifies each research work's titles, reflecting the investigation's primary focus or theme. The titles often indicate specific approaches to botnet detection, such as traffic analysis, user behavior assessment, or advanced machine learning techniques.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

2.2. Methodology

The methodologies employed in these studies reveal the diverse techniques used for botnet detection. These include:
1. Machine Learning Algorithms: Many studies employ machine learning models (e.g., Random Forest, CNN) to classify and detect botnet-related traffic patterns, utilizing historical data to improve predictive accuracy[14], [15], [16], [17], [18], [19], [20]
2. Genetic Algorithms: Some research leverages genetic algorithms to optimize detection parameters, enhancing detection robustness through evolutionary strategies [15].
3. Traffic Behavior Analysis: Various studies analyze traffic behavior, focusing on patterns in TCP/UDP flows or DNS queries to detect anomalies indicative of botnet activity [17], [19], [20].
4. Deep Reinforcement Learning: More advanced methodologies like Deep Reinforcement Learning allow models to adapt dynamically in response to feedback from detection environments, showcasing the ability to combat evolving botnet evasion tactics[19].

2.3. Key Focus

The key focus of each study is centered on specific objectives within botnet detection:
1. Botnet Activity Detection: Many studies aim to identify botnet activity through the detection of abnormal behaviors in network traffic or user interactions[16], [17], [18], [20], [21].
2. Differentiation Between Human Users and Bots: Some research aims to distinguish between human and bot traffic using behavioral analysis, improving the detection of malicious activities[14], [21], [22].
3. Early DDoS Attack Prediction: A subset of studies focuses on predictive models that detect and mitigate DDoS attacks before they occur, emphasizing proactive approaches in cybersecurity[18].

2.4. Challenges Addressed

The challenges addressed by the studies are central to understanding the complexities of botnet detection:
1. Encrypted Traffic: A common challenge lies in handling encrypted communications, which obscure network packets and make malicious traffic detection difficult[15], [16], [17].
2. Scalability: Many studies deal with the difficulty of scaling detection systems to handle large datasets typical in modern networks [15], [16], [18], [20].
3. User-Agent Spoofing: Some research tackles the problem of bots manipulating user-agent strings to impersonate legitimate users, complicating detection [14], [21].

2.5. Results/Performance

The results and performance metrics presented in these studies provide quantitative measures of their success:
1. Accuracy Rates: Most studies report accuracy rates above 90%, showcasing the strong performance of their detection models in identifying botnet activities).
2. Precision and Recall: Some studies report precision and recall metrics in addition to accuracy, reflecting their effectiveness in reducing false positives and correctly identifying true botnet activities.

Study	Methodology	Key Focus	Challenges Addressed	Results/Performance
Smart Approach for Botnet Detection	Machine learning with **Genetic Algorithm** and **traffic behavior analysis**.	Botnet detection through **TCP/UDP traffic**.	- Handling **encrypted traffic**. - Scalability for large networks.	**97% accuracy** in detecting Botnet activities using real datasets.
Detecting Botnet based on Network Traffic	**Machine learning** applied to **DNS traffic analysis**.	Identifying Botnet through abnormal **DNS queries**.	- **Encrypted traffic**. - Large-scale DNS traffic analysis.	**99.64% accuracy** with Random Forest for Botnet detection.

Botnet Detection Based on Traffic Behaviour Analysis and Flow Intervals	Machine learning-based traffic behavior analysis using flow intervals.	Detecting Botnets via behavior analysis of network traffic (TCP/UDP flows).	- Handling encrypted traffic. - Detecting Botnets with minimal traffic.	Achieves high accuracy (~97%) in detecting Botnet activities, even in encrypted traffic.
Bot Detection Model Using User Agent and User Behaviour for Web Log Analysis	Analyses user-agent strings and web browsing behavior in access logs.	Differentiating between human users and Bots in web traffic	- Bots faking user-agent strings. - Similar behavior between Bots and humans.	Accurately distinguishes Bots from humans using web log behavior patterns.
A Distributed Architecture for DDoS Prediction and Bot detection	A hybrid model combining clustering and traffic analysis for early DDoS detection.	Predicting DDoS attacks and identifying Botnet participants.	- Detecting DDoS attacks early. - Handling large-scale network traffic.	99.9% accuracy in detecting Botnets and early DDoS signals using a distributed architecture.
Bot Graph: Web Bot detection Based on Sitemap	It uses CNN to classify web behavior mapped to sitemaps as trace images.	Detecting web Bots based on behavior mapped to website sitemaps.	- Bots bypassing identity checks (IP, user-agent). - Scaling for large websites.	Use a sitemap and CNN approach to achieve ~95% precision and recall for Bot detection.
Web Bot Detection Evasion Using Deep Reinforcement Learning	Reinforcement learning (RL) trains Bots to adapt and evade detection by learning from feedback. Bots modify behavior based on web log responses to avoid detection.	Bot evasion and detection in web logs through adaptive Bot behavior.	- Continuous adaptation of Bots. - Real-time detection and re-training of models.	Bots successfully evaded detection **multiple times**, even after re-training the detection system.
An Analysis of Anomalous User Agent Strings in Network Traffic	**Regular expression-based** anomaly detection is applied to a dataset of 150 billion UA strings.	Detecting anomalies in **User-Agent (UA)** strings to identify malicious activity.	- Detection of rare but significant UA anomalies. - Global distribution of anomalies.	Detected **0.1485%** anomalous UA strings, correlating with malicious activity from **91,000 clients** worldwide.
Detecting Malicious Activities with User-Agent-Based Profiles	**Grammar-guided UA string classification** using context-free grammar (CFG) and heuristics for non-standard UA strings.	Identifying malicious behavior by analyzing **User-Agent (UA)** string anomalies in HTTP traffic.	- Classifying Both standard and non-standard UA strings. - Detecting **spoofed** UA strings used by Bots.	High accuracy in classifying UA strings, **reduced false positives**, and **accurate identification** of malicious activities using UA profiles.

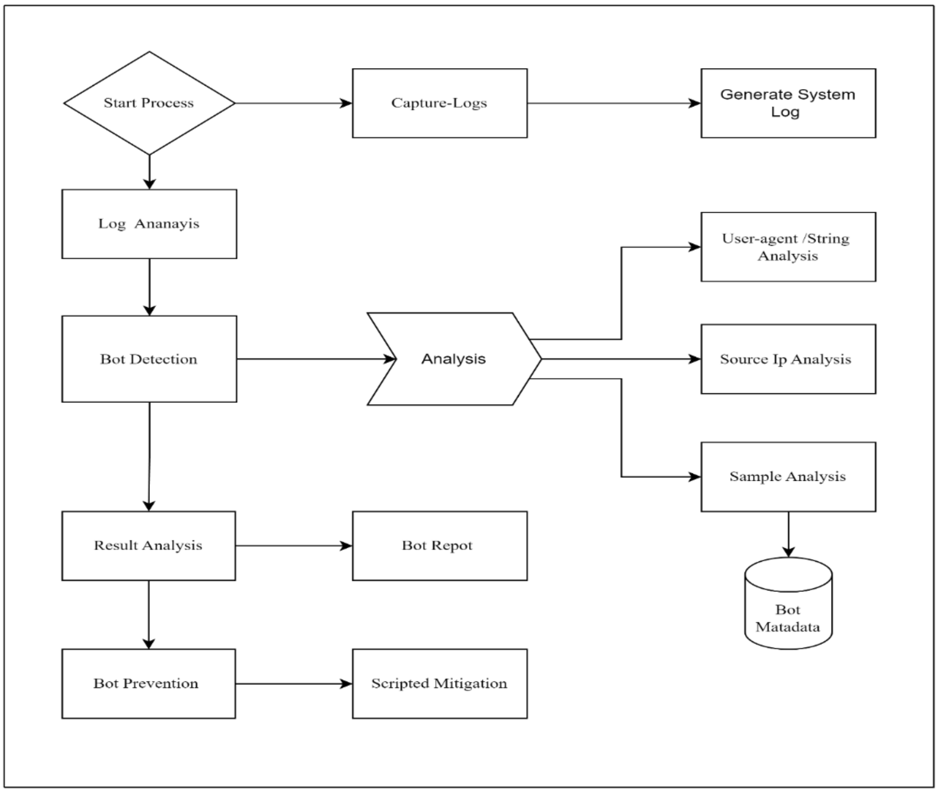
Botnet Detection Techniques and Research Challenges	Combines **honeypot-based** and **IDS** approaches.	Overview of detection techniques for **multiple protocols** (DNS, IRC, HTTP, P2P).	- **Encrypted communication**. - Fast-flux domain evasion.	Discussion-based paper; no specific performance metrics provided

Table 1 Literature survey

III. METHODOLOGY

To implement this methodology, we use Python to create data extraction and analysis and to process scripts for easy data analysis. In this research, we use the PyCharm tool to develop scripts for capturing logs, identifying the Bot, and blocking the Bot via firewall Rule scripts. The Python libraries are Pandas, Skype, Report Lab, matplotlib, and many others libraries.

The three main real-time bot detection and blocking processes are log capture, data analysis, and prevention to detect and block Bots.



Flowchart 1 Methodology

3.1 Log Data Capture

For Bot detection, we use log-based database files. The bot detection data analysis method uses log data in two ways. First is real-time captured log data at the moment of data analysis. These logs captured files can use real-time log analysis, and the second is previous or another system-captured log files.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Real-time log capturing: this type of log will capture system run time log files based on log data like firewall logs, network logs, applications logs, and system log types of logs data will be captured via log capturing files. A real-time log-capturing system will capture this log type via log-capturing techniques.

Another method is using previously captured log files: this type of log can be any log file like system, network, browser, registry, or many kinds of log data files use log data files. This type of log data provides captured log files that detect the Bot in other devices (log data like firewalls and servers) Bot detection via their log files.

3.2 Data Analysis

For the bot detection analysis phase, all data will used as a machine learning data extraction method to extract log data and analyze the database based on data types like [i] user-agent or strings, [ii] IP source address, and [iii] sample analysis. We will follow the three phases below to analyze and identify the bot in the log file and detect it in the log database.
User-agent / string Analysis: the first phase of identifying log data via user-agent or string-based detection in the log database to identify Bot strings or user-agents.
Source IP analysis: The second phase of data identification uses IP sources in log data to analyze data and identify Bot based on the IP source of Bots.
Sample Analysis: The third phase of the analysis database in the pre-defined bot listed database in bot metadata is to compare log data with the sample bot database with Bot Mata Data and analysis bots.

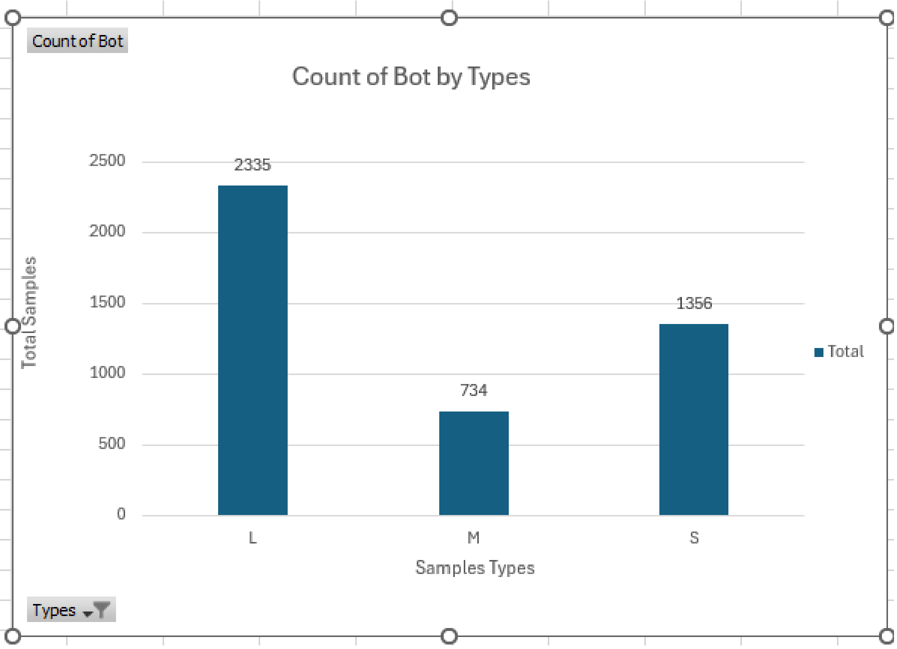


Chart 1 Sample Analysis Database

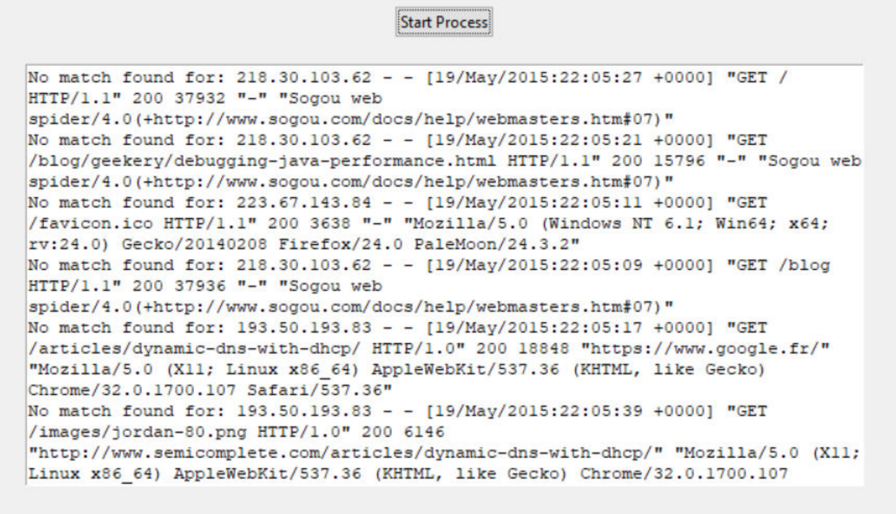
3.3 Result Analysis

The first log data will be captured or uploaded in bot detection for the first data analysis process. When log data is uploaded, the data analysis process starts and extracts all log databases in three categories: first, a list of user-agent or strings; second, IP addresses; and third, a list of other remaining data contents.
User-agent /springs will analyze string database base data to compare and analyze result data.
IP-based data will be analyzed and compared data with IP-based data
Ramming data will be analyzed in extraction and analysis using a machine learning model based on data extraction. Then, each data will be compared with Bot Data samples and the result analysis database. The data analysis will be similar to matching data and giving results in a database.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

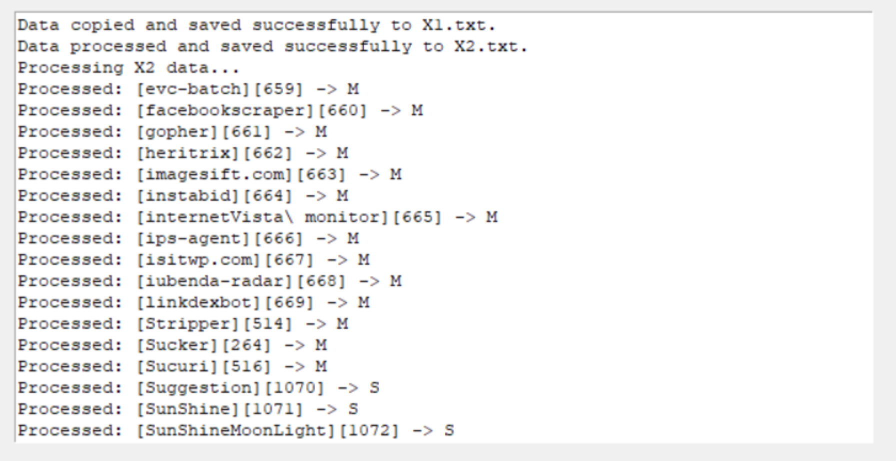
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



```
No match found for: 218.30.103.62 - - [19/May/2015:22:05:27 +0000] "GET /
HTTP/1.1" 200 37932 "-" "Sogou web
spider/4.0(+http://www.sogou.com/docs/help/webmasters.htm#07)"
No match found for: 218.30.103.62 - - [19/May/2015:22:05:21 +0000] "GET
/blog/geekery/debugging-java-performance.html HTTP/1.1" 200 15796 "-" "Sogou web
spider/4.0(+http://www.sogou.com/docs/help/webmasters.htm#07)"
No match found for: 223.67.143.84 - - [19/May/2015:22:05:11 +0000] "GET
/favicon.ico HTTP/1.1" 200 3638 "-" "Mozilla/5.0 (Windows NT 6.1; Win64; x64;
rv:24.0) Gecko/20140208 Firefox/24.0 PaleMoon/24.3.2"
No match found for: 218.30.103.62 - - [19/May/2015:22:05:09 +0000] "GET /blog
HTTP/1.1" 200 37936 "-" "Sogou web
spider/4.0(+http://www.sogou.com/docs/help/webmasters.htm#07)"
No match found for: 193.50.193.83 - - [19/May/2015:22:05:17 +0000] "GET
/articles/dynamic-dns-with-dhcp/ HTTP/1.0" 200 18848 "https://www.google.fr/"
"Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/32.0.1700.107 Safari/537.36"
No match found for: 193.50.193.83 - - [19/May/2015:22:05:39 +0000] "GET
/images/jordan-80.png HTTP/1.0" 200 6146
"http://www.semicomplete.com/articles/dynamic-dns-with-dhcp/" "Mozilla/5.0 (X11;
Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107
```

Figure 2 Data analysis process

After data is funded in mate data, Bot types will be identified. All results of the database Bot will result in both types and give the result that the database of Bot is legitimate or malicious with a defined first letter of the legitimate “L” and malicious “M” or suspicious” S” type of database.



```
Data copied and saved successfully to X1.txt.
Data processed and saved successfully to X2.txt.
Processing X2 data...
Processed: [evc-batch][659] -> M
Processed: [facebookscraper][660] -> M
Processed: [gopher][661] -> M
Processed: [heritrix][662] -> M
Processed: [imagesift.com][663] -> M
Processed: [instabid][664] -> M
Processed: [internetVista\ monitor][665] -> M
Processed: [ips-agent][666] -> M
Processed: [isitwp.com][667] -> M
Processed: [iubenda-radar][668] -> M
Processed: [linkdexbot][669] -> M
Processed: [Stripper][514] -> M
Processed: [Sucker][264] -> M
Processed: [Sucuri][516] -> M
Processed: [Suggestion][1070] -> S
Processed: [SunShine][1071] -> S
Processed: [SunShineMoonLight][1072] -> S
```

Figure 3 resulted in database

it will give a visual table-based result database and a list of Bots funded in the database. It will provide the results on Bots. If the Bot is colorated, the result will be Bots in the “Geen” color list, and if the color is malicious, then provide red type “Red “color result Bots discolored if it is malicious, then offer “Black “color Bot in the table.

It also provides flowchart-based reports with results of percentages based on the listed Bot lists in various log databases with legitimate malicious or suspicious percentage results and with impact based on how it will affect the system's future.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

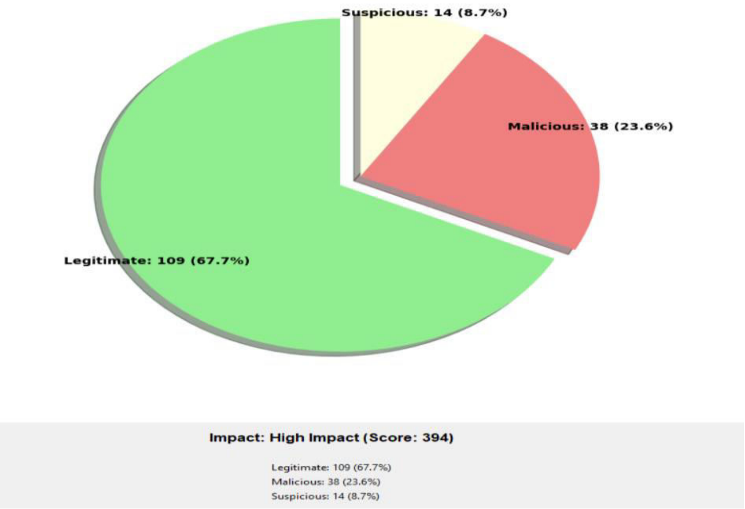
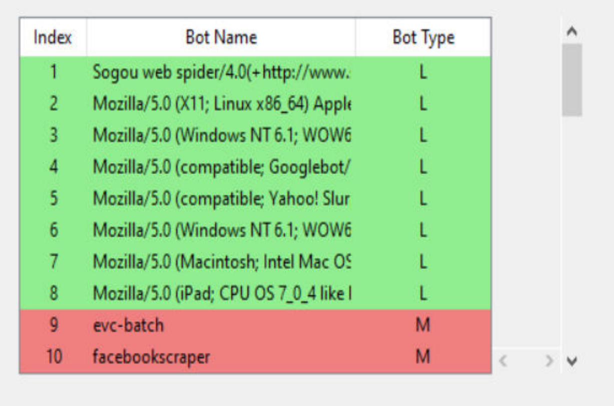


chart 2 Result Impact Chart

It also provides a report with highlighted bot strings and users as agent list log stings and highlighted data in a report that can be identified.

3.4 Prevention Bot

To prevent the system from being malicious or suspicious, we can use some phases to identify the blocks and remove them if they already have them in the stem via the malicious patch script installed. Analysis Bots for analysis Bot need to be blocked for system prevent from Bot in Windows Txt file upload Bot list. Also, check the log file report highlighting the bot with a highlighter and giving more details about the bot.



Index	Bot Name	Bot Type
1	Sogou web spider/4.0(+http://www.	L
2	Mozilla/5.0 (X11; Linux x86_64) Appl	L
3	Mozilla/5.0 (Windows NT 6.1; WOW6	L
4	Mozilla/5.0 (compatible; Googlebot/	L
5	Mozilla/5.0 (compatible; Yahoo! Slur	L
6	Mozilla/5.0 (Windows NT 6.1; WOW6	L
7	Mozilla/5.0 (Macintosh; Intel Mac OS	L
8	Mozilla/5.0 (iPad; CPU OS 7_0_4 like I	L
9	evc-batch	M
10	facebookscraper	M

Table 2 Result logfile bot-data table.

The upper report highlighted Bot would give more reliable information based on Bot behavior.

Bots are measured as two types running in the system: one on the Internet or a network and the other on an OS. For removing network Bots, use a log file that highlights data on log row data to identify that IP source and that used port to block that. If it runs on the system, check that Bot user service or process ID to detect that it has uninstalled the patch file and scripts and removed them after blocking that source IP with that used port. For blocking IP or port, use a python file that uses CMD execution script run to add rule firewall drop source of that Bot address and port and prevent system form that Bots. The bot will be blocked Bot automatically in the system. They are security system rules based on identifying that log string and finding the IP and source code of that log and then finding Bot Data via log string, and it will be

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.625| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

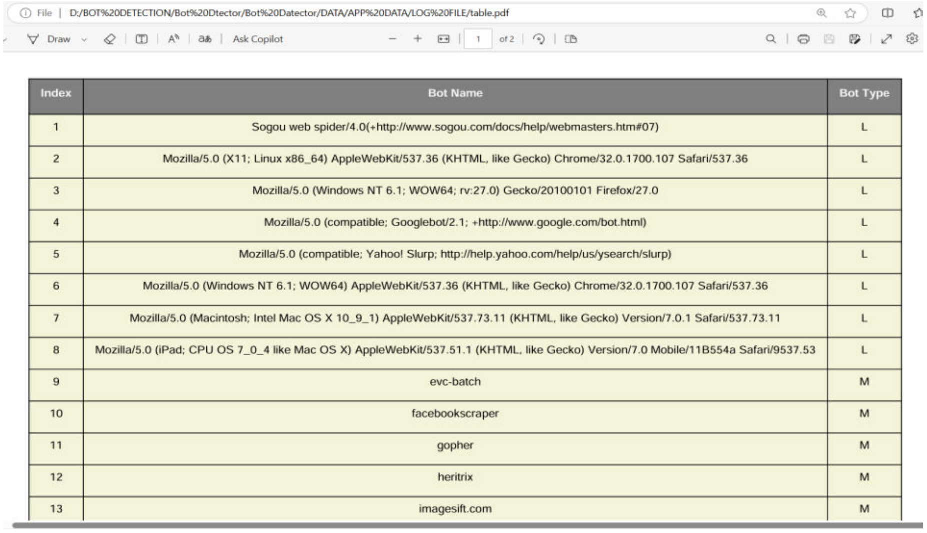
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

blocked port and IP source of that Bot via python script for system firewall rule add. Concluding all the work results, it will successfully identify the Bot in the log database and block it via firewall-scripted CMD execution rules farewells rules.

IV. IMPLEMENTATION

Implementing the bot detection and mitigation methodology is a systematic and multi-faceted approach to safeguard systems from automated threats. This process begins with **capturing network logs**, where critical data points such as user-agent strings, source IP addresses, and other relevant metadata are collected from various system components. This foundational step provides the raw data needed to identify potential bot activity. Once the logs are captured, the next step involves **log analysis**, which utilizes advanced algorithms and machine-learning techniques to scrutinize the collected data. During this analysis, algorithms are employed to detect patterns, anomalies, and irregularities characteristic of bot behavior. For example, a sudden spike in requests from a single IP address or the presence of unusual user-agent strings can indicate bot activity. Upon flagging potential bots through initial analysis, the process advances to **bot detection**, where a more in-depth examination is performed on the identified entities. This includes a comprehensive analysis of user-agent strings to discern any inconsistencies or patterns that might signify malicious intent. Simultaneously, the source IP addresses are evaluated to determine their geographic origin and reputational risk, enabling analysts to categorize the threats based on their severity. In addition to these analyses, samples of the bot activity are collected and examined to gain further insights into their modus operandi, revealing vulnerabilities that can be exploited to thwart future attacks. After this extensive analysis phase, the findings are compiled into a **detailed bot report**. This report serves as a critical resource, outlining the characteristics of each detected bot, including their classification (e.g., legitimate, malicious, or suspicious), behavioral patterns, and potential impacts on the system. Such comprehensive documentation not only aids in understanding the current threat landscape but also assists in formulating strategic responses to mitigate future risks.

To enhance system resilience, the methodology incorporates proactive measures designed to prevent future bot attacks. This includes deploying **automated scripts** programmed to respond in real-time to detected threats, effectively neutralizing any malicious activity before it can inflict damage. These scripts can automatically block suspicious IP addresses, alert system administrators, and implement security protocols to fortify defenses. Using this detection methodology, we will analyze real-time captured log data. It will analyze data and compare Bots. Bot metadata will result in strings or user agents being logged in network logs. It will compare their data, extract that result data, and provide a report regarding Bots. If ant Bot is identified as running on systems and finds anybody's logs or user-agent strings, then report about that with the use-based classified Bot types and impact based on the chart below.


File | D:/BOT%20DETECTION/Bot%20Dtector/Bot%20Datector/DATA/APP%20DATA/LOG%20FILE/table.pdf
Draw
Ask Copilot
1
of 2

Index	Bot Name	Bot Type
1	Sogou web spider/4.0(+http://www.sogou.com/docs/help/webmasters.htm#07)	L
2	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107 Safari/537.36	L
3	Mozilla/5.0 (Windows NT 6.1; WOW64; rv:27.0) Gecko/20100101 Firefox/27.0	L
4	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	L
5	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)	L
6	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.107 Safari/537.36	L
7	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.73.11 (KHTML, like Gecko) Version/7.0.1 Safari/537.73.11	L
8	Mozilla/5.0 (iPad; CPU OS 7_0_4 like Mac OS X) AppleWebKit/537.51.1 (KHTML, like Gecko) Version/7.0 Mobile/11B554a Safari/9537.53	L
9	evc-batch	M
10	facebookscraper	M
11	gopher	M
12	heritrix	M
13	imagesift.com	M

Figure 4 bot list result

Based on the result given, bots are legitimate perspectives, malicious perspectives, or suspicious perspectives. The resulting chat shows some of the bot user-agent strings, and some bots are shown in the result table. This research perspective identifies and blocks running system Bots via Bot detection methodology. This methodology is based on real-time Bot detection and Prevention via firewall rules drop bot sources.

V. CONCLUSION

This research focuses on utilizing machine learning for data extraction and analysis and implementing a log-capturing methodology for system log gathering. A key component of our approach is developing a bot-blocking rule-based method. This methodology includes a real-time log capture system to detect and block malicious bots effectively.

We have established a framework for detecting and mitigating real-time bot activity through our work. Looking ahead, we aim to extend this methodology to larger networks by leveraging machine learning algorithms, which will enhance processing speed and expand the bot detection database for improved analysis.

In future applications, we envision integrating these methodologies with Security Information and Event Management (SIEM) tools. This integration will facilitate comprehensive analyses of botnets and other user-agent behaviors, allowing for the mitigation of malicious user agents and IP sources through advanced script-based firewall rules.

REFERENCES

[1] "11 New Technologies in AI: All Trends of 2023-2024." Accessed: Oct. 08, 2024. [Online]. Available: https://devabit.com/blog/top-11-new-technologies-in-ai-exploring-the-latest-trends/
[2] "DDOS Attacks: What, How, and the Emergence of AI-powered DDoS." Accessed: Oct. 08, 2024. [Online]. Available: https://www.einfochips.com/blog/ddos-attacks-what-how-and-the-emergence-of-ai-powered-ddos/
[3] "What is a bot attack? | Cloudflare." Accessed: Oct. 08, 2024. [Online]. Available: https://www.cloudflare.com/learning/bots/what-is-a-bot-attack/
[4] H. Wang, H. He, W. Zhang, W. Liu, P. Liu, and A. Javadpour, "Using honeypots to model botnet attacks on the internet of medical things," Computers and Electrical Engineering, vol. 102, p. 108212, Sep. 2022, doi: 10.1016/J.COMPELECENG.2022.108212.
[5] "Detecting Suspicious and Malicious Activity on Your Network Network Security Tools." Accessed: Oct. 08, 2024. [Online]. Available: https://www.alertlogic.com/blog/detecting-suspicious-and-malicious-activity-on-your-network/
[6] "What is Network Detection and Response and Why is it So Important?" Accessed: Oct. 08, 2024. [Online]. Available: https://securityintelligence.com/posts/network-detection-and-response-network-security/
[7] "What are bots, and how do they work?" Accessed: Oct. 08, 2024. [Online]. Available: https://www.techtarget.com/whatis/definition/bot-robot
[8] "Bot Detection | How to Detect Bots in 2023 | Radware." Accessed: Oct. 08, 2024. [Online]. Available: https://www.radware.com/cyberpedia/bot-management/bot-detection/
[9] "What is a Bot? - Types of Bots Explained - AWS." Accessed: Oct. 08, 2024. [Online]. Available: https://aws.amazon.com/what-is/bot/
[10] "What is a bot? | Bot definition | Cloudflare." Accessed: Oct. 09, 2024. [Online]. Available: https://www.cloudflare.com/learning/bots/what-is-a-bot/
[11] "Types of Bots: An Overview of Chatbot Diversity | botnerds.com." Accessed: Oct. 08, 2024. [Online]. Available: https://botnerds.com/types-of-bots/
[12] "What is a Bot? - Types of Bots Explained - AWS." Accessed: Oct. 08, 2024. [Online]. Available: https://aws.amazon.com/what-is/bot/
[13] "What is Bot Detection? | Avi Networks." Accessed: Oct. 08, 2024. [Online]. Available: https://avinetworks.com/glossary/bot-detection/
[14] Y. Zhang et al., "Detecting malicious activities with user-agent-based profiles," International Journal of Network Management, vol. 25, no. 5, pp. 306–319, Sep. 2015, doi: 10.1002/NEM.1900.
[15] A. Obeidat and R. Yaqbeh, "Smart Approach for Botnet Detection Based on Network Traffic Analysis," 2022, doi: 10.1155/2022.

[16] N. V. Tuan Hiep, “Detecting Botnet based on Network Traffic,” International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 3, pp. 3010–3014, Jun. 2020, doi: 10.30534/IJATCSE/2020/79932020.

[17] D. Zhao et al., “Botnet detection based on traffic behavior analysis and flow intervals,” Comput Secur, vol. 39, no. PARTA, pp. 2–16, Nov. 2013, doi: 10.1016/J.COSE.2013.04.007.

[18] B. M. Rahal, A. Santos, and M. Nogueira, “A Distributed Architecture for DDoS Prediction and Bot Detection,” IEEE Access, vol. 8, pp. 159756–159772, 2020, doi: 10.1109/ACCESS.2020.3020507.

[19] C. Iliou, T. Kostoulas, T. Tsikrika, V. Katos, S. Vrochidis, and I. Kompatsiaris, “Web Bot Detection Evasion Using Deep Reinforcement Learning,” ACM International Conference Proceeding Series, Aug. 2022, doi: 10.1145/3538969.3538994.

[20] Y. Luo, G. She, P. Cheng, and Y. Xiong, “BotGraph: Web Bot Detection Based on Sitemap,” Mar. 2019, Accessed: Oct. 08, 2024. [Online]. Available: https://arxiv.org/abs/1903.08074v2

[21] T. Tanaka, H. Niibori, S. Li, S. Nomura, H. Kawashima, and K. Tsuda, “Bot Detection Model using User Agent and User Behavior for Web Log Analysis,” Procedia Comput Sci, vol. 176, pp. 1621–1625, Jan. 2020, doi: 10.1016/J.PROCS.2020.09.185.

[22] J. Chen, G. Gou, and G. Xiong, “An analysis of anomalous user agent strings in network traffic,” Proceedings - 21st IEEE International Conference on High-Performance Computing and Communications, 17th IEEE International Conference on Smart City and 5th IEEE International Conference on Data Science and Systems, HPCC/SmartCity/DSS 2019, pp. 1771–1778, Aug. 2019, doi: 10.1109/HPCC/SmartCity/DSS.2019.00243.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

☎ 9940 572 462 6381 907 438 ✉ ijircce@gmail.com

www.ijircce.com

Scan to save the contact details