International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
Article Received: 28 July 2022 Revised: 05 September 2022 Accepted: 12 September 2022

Central Pivot Heuristics for Botnet Attack Defense in Iot

***1G. Chandana Swathi, 2G. Kishor Kumar, 3 A.P. Siva Kumar**

*1Research Scholar, Department of Computer Science and Engineering, JNTUACE, Ananthapuramu, A.P, India.
*1Email: chandanaswathisura@gmail.com
2Associate Professor, Department of Computer Science and Engineering, RGMCET, Nanadyal, A.P, India.
2Email: kishorgulla@yahoo.co.in
3Assistant Professor, Department of Computer Science and Engineering, JNTUACE, Ananthapuramu, A.P, India.
3Email: sivakumar.ap@gmail.com
*Corresponding Author

Abstract: Botnet assaults on IoT systems have become a big issue, and several strategies for botnet protection have been investigated by the academic and industry communities. While many of these methods are practical and effective for botnet attack prevention, one of the important limits is the load factor on the servers that manage monitoring and control in addition to catering to client system requests. To address load factor difficulties, the focus of this study report is on the conditions of installing a four-layer security control system based on the notion of central pivot points. Inspired by the effective and systematic Markov Chains concept, this publication proposes a four-layer filtering model that shows if botnet detection and prevention methods for servers are required. The model's simulated experimental study demonstrates the potential scope of deploying the system. The study also highlights the future possibilities of model improvisation that can reduce any erroneous signal production that is judged necessary.

Keywords: Markov Chains, botnet attacks, virtual machine monitor, Central Pivot Range- Server Traffic Detection Model.

1 Introduction

The good and bad of technological improvements have always posed a challenge to the most cutting-edge developments in the realm of information technology. The growth of Internet of Things (IoT) technologies has altered the paradigm of device connectivity and control. However, in terms of the negative impact, the threat of botnet concerns has become a big challenge.

There is a separate set of concerns that impair service quality in the IoT context, which are influenced by botnet conditions. Initially, botnets were small-scale, with a small number of servers or systems under attack. However, in recent years, it has been typical for botnets to influence thousands of devices, and more specifically, one botnet can manage the number of sub-systems with malware. According to Time-Frame, the scope of million-plus-size botnets is not implausible, and the impact of such advances is enormous [1].

Botnets, as a systematic activity, can have an impact on users in both direct and indirect ways. In an illustrative situation, the infected machine is no longer under the control of the legitimate user. Given the sensitive information held in information systems, data privacy control compliance, and other such characteristics, it is critical to protect the systems from botnet attacks [2].

The impact of a botnet is twofold: first, the susceptible system is affected and under the control of the hacker; second, the hackers will use the system as a bot under the command system to conduct assaults on other external systems. When an organization's system network is attacked by botnet hackers, it can result in issues such as losing control of the servers and affecting the organization's business continuity activities.

Some of the common challenges imperative in the case of the botnets are.

- Launching of DDoS attacks over the other websites or services
- Distributing spam emails or malware into other systems
- Unlawful mining of digital currencies
- If the target systems do not have adequate prep for handling the attacks, such systems too can face harm [3].

Industry players must work together to address the issue. Many industrial and academic research projects have focused on the subject of internet service-related challenges, and some modern solutions for securing systems from botnet attacks are widely employed. Despite the fact that several major solutions have emerged in the process, the fundamental condition of prevention being better than cure continues to play a vital role in the mitigation of botnet attacks. As a result, it is prudent for server monitoring teams to rely on advanced technologies that can assist in understanding the increasing traction and performance of the systems to session requests and how the steps required to target botnet attacks may be efficiently modified [4].

There is an inherent load factor on the servers as the programs operate in the background for detection among the rising innovations of deploying various types of solutions for botnet detection. This could have an impact on the server's optimal resource use. As a result, using a lighter screening layer on the servers can aid in the final monitoring and deployment of the prevention systems as needed. Such a procedure will improve the server services' overall operational quality [2].

In accordance with the aforementioned goal of improving the early-stage indicators that refer to the performance of the systems in terms of indicating the performance, this manuscript proposes a modern model of adapting the central pivot points and the support and resistance zones that can help the server administration controls make informed decisions on the system's performance. Fundamentally, the suggested system can function as a dual-edge solution in that the effective performance conditions of the server network can be monitored, and any decline in performance indicated below the pivot point can be used to warn of a potential threat to the application systems.

The remaining sections of the manuscript are organized as follows: Section 2 delves into related research on botnet detection algorithms for IoT networks. Section 3 delves deeper into the materials and procedures employed in the proposed contribution. Section 4 depicted the experimental investigation and performance analysis of the proposed approach, which scaled when compared to other contemporaneous models. Finally, section 5 summarized the contribution, performance evaluation results, and potential future studies. The final part contains a list of all the references used in the manuscript.

2 Related Work

The authors of the paper [5] examined the scope of using machine learning models for the detection of DoS attacks. The authors of the paper address the importance of modifying HTTP/2 web servers, focusing on the volume, intensity, and rising stress on network bandwidths. The study's approach of relying on HTTP/2 services was examined using four machine learning classifiers, such as the NB, SVM, JRip, and Decision Tree, as well as stealthy traffic features, which are illustrated with higher percentages of false alarms. The study's findings are said to be a potential model for addressing the issue and averting repeated attacks on traffic models as a result of service outages [5].

The authors of a recent model addressed in [6] focused on a vital situation at hand, where the requirement for early detection of attacks is imminent. The authors of the paper suggested a time-sequential model in which malware-generated network attacks are anticipated using time-sensitive variables, focusing on the unique elements of how early detections might be notified. According to the study, there is a condition in which the Markov chain technique is employed to estimate the conditions of real harmful traffic. The paper also mentions the use of the semi-Markov Chain model to estimate the breadth and time of malware attacks, allowing controllers to make informed decisions.

The authors of the paper [7] analysed the breadth of attacks on the kernel of information systems and provided a model for blocking active data kernel rootkit assaults to monitor or observe kernel memory access based on VMM (Virtual-machine monitor) regulations. Regardless of whether VMM was introduced as an external or outsider monitoring system, it detects changes in kernel data reports and generates the necessary alarms.

The study claims to have significant performance outcomes in the process, as well as those that can aid in the development of long-term outcomes in the process of protecting kernels against attacks [7].

In [8,] the study's authors provided a scenario in which the server's performance parameters are taken into account, and as a result, a unique model of a sequential architecture-based support system is proposed, which can aid in enhancing botnet identification based on a sequential pattern. The study promises considerable confirmation in the detection process and can assist enterprises in boosting detection timing and overall solution quality. The essence of the concept, however, is profoundly about the characteristics chosen for execution and how solutions might be more pragmatic.

The study's authors [9] examine the extent of a system that can be implemented at both the client and host ends. Based on the IP fluxing conditions, the model is mainly focused on botnet communication traffic arising in HTTP, P2P, IRC, and DNS. HANABot is the model's proposed algorithm that works in terms of preprocessing and feature extraction to observe the variation in botnet activity and genuine service behaviour.

One of the most critical constraints in botnet detection models is the question of how antivirus solutions target malware detection, as well as the limitations of such solutions, given the constant creation of new types of attacks. As a result, the model emphasises an event-driven approach that can be changed to solve the limits. However, the issue with such models is dealing with the consequences that arise as well as the necessary countermeasures. The study's discussion of modern event-driven analysis models is alluded to as a potential solution that can improve the overall process outcome [10].

The study's authors [11] discussed the model of optimised extreme gradient boosting and feature selection model for dealing with botnet attack detection in its early phases. The model proposed in the study is based on the Fisher-Score centric feature selection model, in which genetic-based extreme gradient boosting models are used for feature determination and discarding irrelevant features with the goal of minimising intra-class distances and maximising inter-class distances. The study's authors describe the method as a promising system for observing botnet attacks in the early timeframes.

A study [12] examines the extent of CART algorithm-based machine learning solutions that can aid in botnet detection. The authors' experimental analysis is based on publicly available IDS datasets, and the results refer to the effective outcome of the model over the Bayer's classifier, as well as overall detection effectiveness with CART, in comparison to other similar solutions.

A recent literature analysis [13] is primarily concerned with the taxonomy of botnets, the history of botnet attacks, and the many types of solutions presented for managing botnet attacks on servers. The model review refers to system gaps, metrics critical for identifying practises, solutions, and algorithms deemed pragmatic for addressing the process. A review of the system aids in the improvement of overall solutions as well as the awareness of the range of consequences to be addressed in managing botnet-related difficulties.

In [14], the survey has studied the breadth of machine learning and other significant models that can be adopted in the process of selecting the suitable solutions for dealing with the threat of botnet assaults.

As an overview of the relevant study, it is clear that many pragmatic ways and technologies for managing botnet attacks are significant and resourceful, as stated. However, one of the major concerns is the lack of any visual representation-centric models that trigger alerts for any human decisions connected to controlling the excessive traffic that is coming to the servers. While machine learning models and other related aspects can assist companies in improving overall botnet prevention models for servers, the difficulty of visual indicators for botnet-related traffic circumstances still has room for new models and solutions.

Transfer Learning Algorithm Intrusion Detection System (TLA-IDS) [15] and Specification Heuristics based Intrusion Detection System (SH-IDS) [16] are the models compared in this paper. These models were employed to deal with the dimensionality issue in the projected values for the characteristics of the provided training corpus. Nonetheless, the models fall into the same category as other existing techniques that try to learn from a corpus of training data that does not portray the dimensionality problem.

The goal of this paper was to find the best specification model for improving the distributed model of intrusion detection in IoT networks constituted by LLN nodes.

3 CPH Model

CPH refers to the model specified as "Central Pivot range-based Heuristics for Botnet Attack Detection across IoT Networks," in which the system performance indicators are basically monitored using the central pivot range system. For in-depth analysis, the model's inspiration, patterns, and process flow model formulation are all discussed in this subsection.

3.1 Scope of Application

In the instance of botnet attacks, botnet systems attack a server and degrade its performance. More often than not, the scope of effect is in the form of DoS or DDoS attacks, which have an influence on the server's performance in handling genuine requests. For example, when a server VM-1 responds to client session requests, there is a potential for a sequential pattern in which the sessions may have peak and non-peak periods, and the servers process the requests accordingly [3], [5].

In normal circumstances, when session requests are made, there is a trend and a pattern to how the requests are processed. However, in the case of excessive traffic, the request processing conditions, bandwidth consumption, and delays abruptly change. In an exemplary situation, if the peak period requests are around 90 requests per second and the

DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

processing ability is working at 85 requests per second, the emergence of abnormal conditions of 400 requests per second indicates some concerns, unless it is a previous event. In many of the previous modelled event-driven machine learning or non-machine learning models, the logic of a sudden surge in session requests or a sudden decline in processing lawful requests works as an incursion affects the servers.

Considering the make span conditions, bandwidth usage, or request processing capacity, or all three in combination, as potential indicators for identifying legitimate requests or processing is the rationale for focusing on the aforementioned requirements. As a result, the model provided in this publication might be used in isolation to watch server performance or as the surface layer for many current solutions in the literature to detect and prevent botnet attacks.

Another significant benefit of the architecture is that it improves the performance of application systems and servers. For example, if the servers are loaded with a greater number of active tracking or monitoring applications operating in the background, the load factor on the servers may be increased, affecting the system's efficacy. As a result, the suggested model can be run as a 4-tier screening model, with the preventive and detection systems triggered if the screening system triggers or warrants an alert.

3.2 Attack Model

Markov Chains are the statistically modeled random process, wherein it is applied vividly over distinct domains, ranging from the next generation to weather analysis, financial modeling, and the probabilities assessment across mathematical scenarios. Usually, Markov chains have been quite intuitive conceptually, and have been accessible, where they would be applied without utilizing any upgraded mathematical or statistical concepts.

For the proposed solution, the inspiration stands in the application of the Markov Chain model in [6], to capture the sequences of malware infection sequences. The following figure 1 representation depicted in [6] refers to the conditions wherein the states of transition diagram modeling botnet infection stages are depicted. Focusing on the various stages and interpretation of the same to the Markov Chain model, the fundamental logic applied to the proposed model is that there is $'t'$ the time that is pragmatic for a shift from stage to stage B. Whereas, if the movement is much quicker, there is a sudden spurt or drop in the performance, based on the assessed metrics.

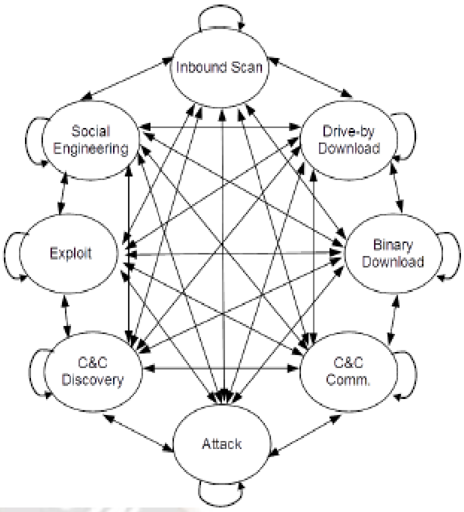


Figure 1: Markov Chain modeled representation of botnet attack stages [6]

As a result, the process of authentic decisions on detections, as well as prevention approaches as needed, shall be introduced to the system, which can aid in the overall process improvement [17].

The reasons for using the Markov Chain model are related to the request's reception and processing capabilities. In a normal situation, the servers could switch between multiple stages ranging from non-peak to peak and very peak circumstances, as well as the return process, which begins with very peak and gradually reduces to peak and towards non-peak. If there are any major changes from very peak to non-peak answers, or a quick surge of requests from non-peak to very peak conditions, such conditions must be cut in the process. The narration under discussion is shown in detail in Figure 2 [18], [19], [20].

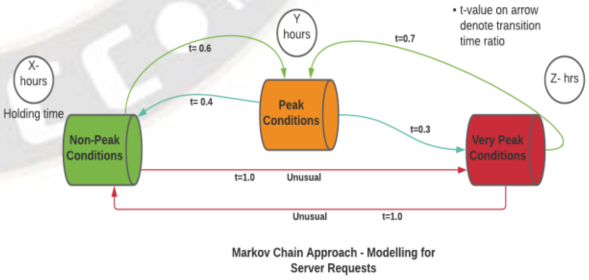


Figure 2: Markov Chain logic discussed

Based on the process of Markov Chain logic discussed in Figure 2, the following set of indicator models is developed to rationalize the structure and the processes imperative for the system alerts essential to address the issue.

3.3 The data

Set-up of attributes in a sequence is the critical step in handling the IoT network transaction. Novel pivot ranges-based scale has projected to signify the botnet alert with minimal false alarming in IoT networks.

The proposal discovers feature attributes having significant diversity in corresponding values projected for chosen records labeled, respectively. In furtherance, it is depicted as sequential patterns of variable size of n. Further filters the sequential patterns of size n that are optimal to both labels positive (prone to a botnet) and negative (benign transactions). The process adopted for suitable features identification are explored in sections followed.

3.4 The selection of attributes correlated to the labels

Dataset D shall be used for notion for network transactions, with clear demarcation of positive labels as D^+ and negative labels mentioned as D^-. In furtherance all the set of attributes depicting the values for network transaction are grouped as a set following Eq 1:

$$Atrb = \left\{a_1, a_2, a_3, ., ., ., a_{|Atrb|}\right\} \text{ ...(Eq 1)}$$

Vectors V labeled as $V_a{}^+ and\ V_a{}^-$ (for positive and negative respectively) are clustered for every attribute $\left\{a \exists a \in Atrb\right\}$. Also, the diversity in the distribution is identified among the values defined as $V_a{}^+ and\ V_a{}^-$, using the Dice similarity coefficient values approach. The model of dice similarity is chosen as it enables easier identification of a different set of values among the same or distinct distribution records [21], [22], [23] and in the proposed solution, it is used for classifying the records as botnet attack prone (positive) or not (negative).

The diversity between two vectors shall assess by Euclidian Similarity Coefficient usc, which is estimated using Eq 2:

$$usc = \frac{2^* | v_a^+ \bigcap v_a^- |}{| v_a^+ | + | v_a^- |} \text{ ...(Eq 2)}$$

The narrative of the equation is as follows

Notations of vectors $\left|v_a^+\right|, \left|v_a^-\right|$ refer to the cardinalities pertaining to the respective vectors and the notation $| v_a^+ \bigcap v_a^- |$ refers to the intersecting values of the chosen vectors.

If the value dsc is lesser than compared to Euclidian similarity coefficient threshold $usct$ (in general $0.7 \le usct < 1$), in such instances two vectors are considered distinct, and the corresponding attribute confines as optimal. The feature selection process is detailed in the further section.

3.5 The features Selection

Identify all the n-grams (sequential patterns of size n) of optimal attributes and pool them as set SPA. Further, sort these n-grams by the size (n value) in a descending format.

For every n-gram $\{pa \exists pa \in SPA\}$ in set SPA

// Begin

// Begin

Set map pvc_{pa}^+ constitute the confidence essential for every sequential pattern value pv of pertaining to sequential pattern attribute pa, as the positive records D^+

- ○ Set PV_{pa}^+ constitutes unique sequential patterns related to values of D^+ positive label in concurrence to the sequential pattern of attributes pa
- ○ Set map pvc_{pa}^- constitute the confidence essential for every sequential pattern value pv of pertaining to sequential pattern attribute pa, as the positive records D^-
- ○ Set PV_{pa}^- constitutes unique sequential patterns related to values of D^- positive label in concurrence to the sequential pattern of attributes pa
- ○ For every record comprising of the set $\left\{r \exists r \in D^+\right\}$ Begin
  - ▪ Identify the values projected for pa in record r pertaining to the sequential pattern of values pv
  - ▪ $if\left(pv \notin PV_{pa}^+\right)$ Begin //In the instances of sequential pattern value pv not being existent over the set PV_{pa}^+
    - $PV_{pa}^+ \leftarrow pv$
    - $pvc_{pa}^+\{pv\} = 1$ // Trigger procedure for initiating the confidence of the sequential pattern values with 1

```
        ▪ End
        ▪ Else Begin //In instance of observing sequential pattern value $pv$ existent in the set $PV_{pa}^{+}$
            • $pvc_{pa}^{+}\{pv\}+=1$ // add count for the confidence of the sequential pattern values with 1
          ▪ End
  o End
  o For every record of the set $\{r \exists r \in D^{-}\}$ Begin
      ▪ Identify the values projected for $pa$ in record $r$ pertaining to a sequential pattern of values $pv$
      ▪ $if\left(pv \notin PV_{pa}^{-}\right)$ Begin // In the instances of sequential pattern value $pv$ not being existent over the set $PV_{pa}^{-}$
          • $PV_{pa}^{-} \leftarrow pv$
          • $pvc_{pa}^{-}\{pv\}=1$ // Trigger procedure for initiating the confidence of the sequential pattern values with 1
      ▪ End
      ▪ Else Begin //If sequential pattern value $pv$ exists in the set $PV_{pa}^{-}$
          • $pvc_{pa}^{-}\{pv\}+=1$ // increment the confidence of the sequential pattern values with 1
        ▪ End
  o End
• End
```

The outcome attained from the process for the sequential pattern of values are depicted as features for further application in the model.

3.6 Detection and Defense Model

Model proposed in this manuscript is completely reliant on the structure of Pivot points analysis.

For any of the request-flow-based metrics can be considered as metrics in silos or in combination when multiple confirmations are essential for analysis.

To the specific metric considered, the winded min, max, and final values are calculated on a given time interval.

Based on the min, max, and final values, the pivot point, resistance stages ($R_{\min}, R_{\max}$) and Support stages (S_1, S_2), and the central pivot ranges are assessed. (The pivot estimation points formulae are mentioned in the respective sections)

Based on the pivot levels drawn for the model, the initial, final, max, and min movement for the respective metric are estimated as per the timeframe considered. If the movement for the metric is within the range of $R_{\min}$ to $S_{\min}$, the request load seems to be normal in function, and if there is any significant variations wherein the metric touches $R_{\max}$ or $S_{\max}$, then the flow of requests said to be prone to denial of service attack, which tends to alert the administration team for further decisive actions [17].

As a reconfirmation of the system, an exponential moving average model shall be used for verification process, wherein if the metric is within the nearby region of Simple moving Average(SMA), then the metric is considered as normal. Whereas, if there is too max or drop attributing to the SMA line, then the process could be interpreted as possibly affected.

3.6.1 Pivot Points

Technically, Pivot point can be referred as a technical calculation that is resourceful for determining the overall trend over distinct time frames for a metric. Profoundly used in the financial and securities exchange domains, the model is highly significant to understand the pivot values for a chosen metric, alongside defining the support and resistance for the metric, in comparison to its current value [14].

The objective of developing the pivot point is to understand the scope of movement or fluctuation that can be estimated, and accordingly the necessary decision making can be pragmatic from the solutions. The estimation model for the pivot points has been explored in following description.

DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

$findPivots(mV)$ Begin	// function that receives a set of values required to identify the pivot points
$\mu=\frac{1}{\|mV\|}\sum_{i=0}^{\|\|mV\|}\left\{e_i\exists e_i\in mV\right\}$	// mean of the vector mV
$\delta=\frac{1}{\|mV\|}\sum_{i=0}^{\|\|mV\|}\left\{\sqrt{(\mu-e_i)^2}\exists e_i\in mV\right\}$	// deviation of the vector mV
$P=\frac{1}{3}\left(e_{\min}+e_{\max}+e_{\|mV\|}\right)-\delta$	//pivot of the vector mV , which is the absolute difference between the mean of the min max and final values of the vector mV and respective deviation of the corresponding vector mV
$R_{\min}=\left\{\log_P(x)\exists p=10,x=100\right\}-\left\{e_{\min}\exists e_{\min}\in mV\right\}$	// min resistance $R_{\min}$ of the pivot P of the vector mV
$R_{\max}=\left\{\begin{matrix}\left\{\log_P(x)\exists p=10,x=10\right\}+\\ \left\{e_{\max}-e_{\min}\exists\left[e_{\min},e_{\max}\right]\in mV\right\}\end{matrix}\right\}$	//maximum resistance point $R_{\max}$ of the pivot P of the vector mV
$S_{\min}=\left\{\log_P(x)\exists p=10,x=100\right\}-\left\{e_{\max}\exists e_{\max}\in mV\right\}$	//minimum support $S_{\min}$ of the pivot point P of the vector mV
$S_{\max}=\left\{\begin{matrix}\left\{\log_P(x)\exists p=10,x=10\right\}-\\ \left\{e_{\max}-e_{\min}\exists\left[e_{\min},e_{\max}\right]\in mV\right\}\end{matrix}\right\}$	//maximum support $S_{\max}$ of the pivot point P of the vector mV
End	//of the function $findPivots(mV)$

3.6.2 Central Pivot Range (CPR)

Central Pivot Range is a versatile statistical computation that comprises 3 level calculation, as central pivot point, top central levels denoted as TC and the bottom central levels denoted with notation BC .

The estimation of CPR is calculated as follows.

$\mu=\frac{1}{\|mV\|}\sum_{i=0}^{\|\|mV\|}\left\{e_i\exists e_i\in mV\right\}$	// mean of the vector mV
$\delta=\frac{1}{\|mV\|}\sum_{i=0}^{\|\|mV\|}\left\{\sqrt{(\mu-e_i)^2}\exists e_i\in mV\right\}$	// deviation of the vector mV
$P=\frac{1}{3}\left(e_{\min}+e_{\max}+e_{\|mV\|}\right)-\delta$	//pivot of the vector mV , which is the absolute difference between the mean of the min max and final values of the vector mV and respective deviation of the corresponding vector mV
$BC=round(\log_{10}3,1)*\left(e_{\min}+e_{\max}\right)-\delta$	// bottom central levels denoted as BC
$TC=\left(\log_{10}100*P\right)-BC$	// top central levels denoted as TC

In a conventional estimation of the CPR, only three variables as Max, min, and final are used for estimating all the three levels. Fundamental objective of using the CPR is to track the performance range of the metric in the previous Time-Frame, and the scope of performance range for the current Time-Frame can be estimated. The CPR values estimated from the earlier times shall remain constant for the current Time-Frame, and there shall not be any dynamic changes to the CRP range on the inter-time period [14].

DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

If the CPR ranges of the metric are on gradual uptrend on downtrend on periodical basis, the system can be seen as smooth functioning, and any abnormal spikes or drops in the CPR range can be seen as reversal or affect in the system performance. Despite that the CPR model is introduced for the securities trading by Mark Fisher, the model can be significant if applied in cross-domain functionalities too.

3.6.3 Simple Moving Average (SMA)

The simple moving average is a statistical estimation of mean value for a chosen time frame. The moving average for a metric is estimated on a progressive daily basis. For the proposed model, the moving average is considered for 20 period, as the 21 is a Fibonacci number, wherein the movements can be expected to have retracements. Detailed brief of the Fibonacci ratios and numbers can be reviewed from [17]. Depending on the load factor, and the customization essential, the moving average periods can be altered to the application as required.

3.6.4 Markov's Chain Estimation

For the chosen metric, the t-value is estimated based on the following formulae.

A transition matrix: The t-value of the $P\left(P_t\right)$ Markov chain $\{X\}$ at time t is a matrix referring to the probability of transitioning among various stages. The estimation of P value is denoted below Eq 3.

$$\left(P_t\right)_{[i,j]} = \mathrm{P}\left(X_{t+1} = j \mid X_t = i\right) \text{ ...(Eq 3)}$$

3.7 Process Flow

Let the following be notional representation of the distinct set of variables chosen in the model.

The notation M be the metric chosen for tracking the performance of the servers.

h *//the max value of the metric M on the previous time interval.*

l *//the lowest value of the metric M on the previous time interval.*

c *//the closing value of the metric M on the previous time interval.*

s *// represents the moving average value for given time interval.*

P_t *//value refers to the Markov's Chain value estimation for the time interval.*

Estimation of CPR & Pivot Ranges (stated in section 3.6.1 and 3.6.2) $P, R_{\max}, R_{\min}, S_{\max}, S_{\min}, BC, and\, TC$

The current consumption value of the metric M, on a running basis is denotes with notation M_{cpv}, based on the closing value or initial or min value according to the chosen metric.

3.7.1 Assessment Process

Stage-1 Authentication - Wider CPR or Narrow CPR

{

The difference in values between Markov's chain value P, top central level TC, and bottom central level BC is narrow, then the volatility is observed to be higher, and there is greater flexibility in the movement values of the metric M.

If the difference in values between Markov's chain value P, top central level TC, and bottom central level BC are wider, then the volatility in the scenario is observed to be less, and the performance of the servers are within the desired lines".

}

Stage-2 - Progressing Pattern Analysis

The state denotes as normal: Eq 4

$$if\left(\left(M_{cpv} > TC\right) \wedge \left(M_{cpv} < R_{\min}\right)\right) \vee \left(\left(M_{cpv} > R_{\max}\right) \wedge \left(M_{cpv} < BC\right) \wedge \left(\mathrm{M}_{cpv} < \mathrm{S}_{\min}\right)\right)$$

...(Eq 4)

The state denotes as botnet alert: Eq 5

$$if\left(\left(\left(M_{cpv} > R_{\min}\right) \wedge \left(M_{cpv} < R_{\max}\right)\right) \vee \left(\left(M_{cpv} > S_{\min}\right) \wedge \left(M_{cpv} < S_{\max}\right)\right)\right)$$

...(Eq 5)

The state denotes as botnet alert: Eq 6

$$if\left(\left(M_{cpv} > S_{\max}\right) \vee \left(M_{cpv} > R_{\max}\right)\right) \text{ ...(Eq 6)}$$

Stage-3: SMA verification

If M_{cpv} value being within the nearby range of SMA line, then the conditions could be considered as intermittent traffic to the system, and necessary load management can be tracked.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

Else if M_{cpv} value being abnormally max or min than the SMA line, then deploy detection systems for the possible botnet attacks investigation and refer to the third stage confirmation.

Stage-4 Verification of Pt Value to M_{cpv} value Eq 7

The state denotes as botnet alert:

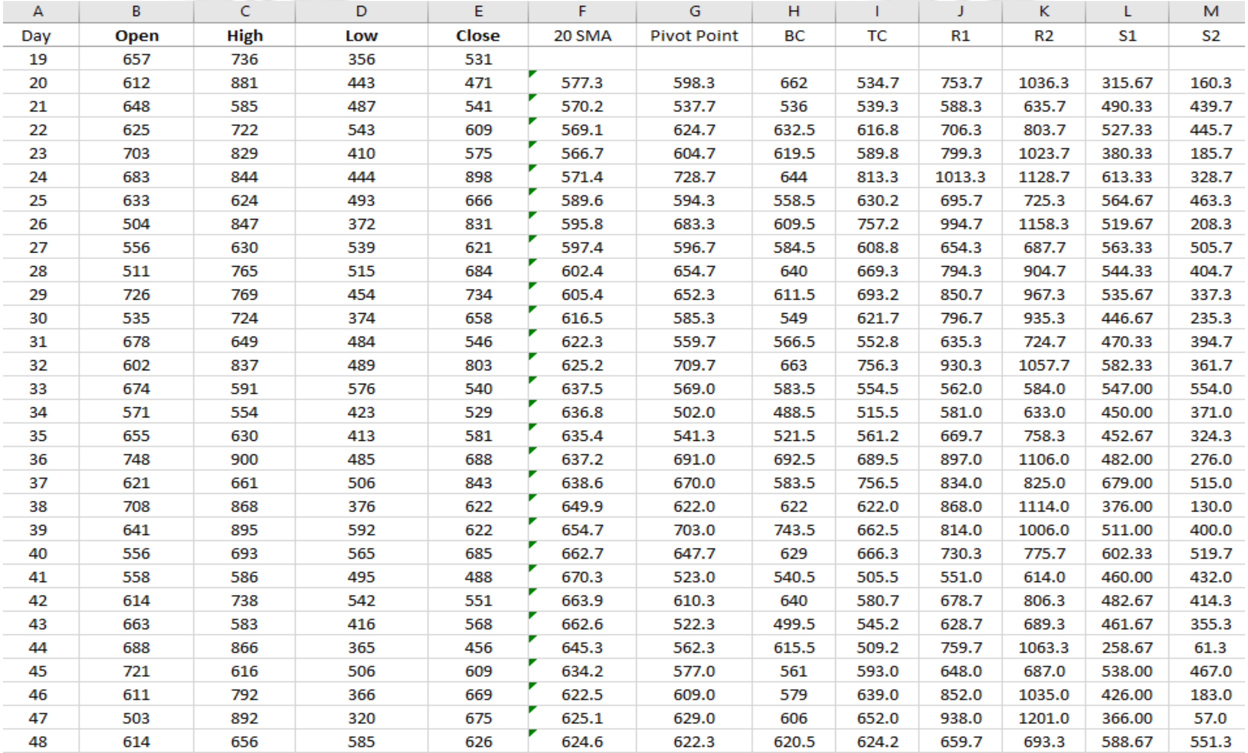
$$if\left(\left(\sqrt{\left(M_{cpv}-P_t\right)^2}\right)>\left(5*10^{\log\frac{1}{100}}\right)\right) \quad \text{...(Eq 7)}$$

4 Experimental Analysis

The proposed model is experimented over a simulated environment structure, wherein the model refers to the conditions of a possible server environment. Focusing on the bandwidth consumption value as a metric, for a server VM-1.

Using the tabulated data of 56 Time-Frames of activity for a server, the SMA, Pivot, TC, BC, R1, R2, S2, S1, conditions are assessed. Accordingly, the following Table 1 representations indicating the Pivot points for the 49th time frame is detailed below. To estimate the ongoing trend for 49th Time-Frame instance in a dataset, the pivot points, support, and resistance must be calculated based on the details from the prior Time-Frame.

Table 1: The information, assessed in the spread sheet, the CPR conditions for the 47th Time-Frame, and 48th time frame



A	B	C	D	E	F	G	H	I	J	K	L	M
Day	**Open**	**High**	**Low**	**Close**	20 SMA	Pivot Point	BC	TC	R1	R2	S1	S2
19	657	736	356	531								
20	612	881	443	471	577.3	598.3	662	534.7	753.7	1036.3	315.67	160.3
21	648	585	487	541	570.2	537.7	536	539.3	588.3	635.7	490.33	439.7
22	625	722	543	609	569.1	624.7	632.5	616.8	706.3	803.7	527.33	445.7
23	703	829	410	575	566.7	604.7	619.5	589.8	799.3	1023.7	380.33	185.7
24	683	844	444	898	571.4	728.7	644	813.3	1013.3	1128.7	613.33	328.7
25	633	624	493	666	589.6	594.3	558.5	630.2	695.7	725.3	564.67	463.3
26	504	847	372	831	595.8	683.3	609.5	757.2	994.7	1158.3	519.67	208.3
27	556	630	539	621	597.4	596.7	584.5	608.8	654.3	687.7	563.33	505.7
28	511	765	515	684	602.4	654.7	640	669.3	794.3	904.7	544.33	404.7
29	726	769	454	734	605.4	652.3	611.5	693.2	850.7	967.3	535.67	337.3
30	535	724	374	658	616.5	585.3	549	621.7	796.7	935.3	446.67	235.3
31	678	649	484	546	622.3	559.7	566.5	552.8	635.3	724.7	470.33	394.7
32	602	837	489	803	625.2	709.7	663	756.3	930.3	1057.7	582.33	361.7
33	674	591	576	540	637.5	569.0	583.5	554.5	562.0	584.0	547.00	554.0
34	571	554	423	529	636.8	502.0	488.5	515.5	581.0	633.0	450.00	371.0
35	655	630	413	581	635.4	541.3	521.5	561.2	669.7	758.3	452.67	324.3
36	748	900	485	688	637.2	691.0	692.5	689.5	897.0	1106.0	482.00	276.0
37	621	661	506	843	638.6	670.0	583.5	756.5	834.0	825.0	679.00	515.0
38	708	868	376	622	649.9	622.0	622	622.0	868.0	1114.0	376.00	130.0
39	641	895	592	622	654.7	703.0	743.5	662.5	814.0	1006.0	511.00	400.0
40	556	693	565	685	662.7	647.7	629	666.3	730.3	775.7	602.33	519.7
41	558	586	495	488	670.3	523.0	540.5	505.5	551.0	614.0	460.00	432.0
42	614	738	542	551	663.9	610.3	640	580.7	678.7	806.3	482.67	414.3
43	663	583	416	568	662.6	522.3	499.5	545.2	628.7	689.3	461.67	355.3
44	688	866	365	456	645.3	562.3	615.5	509.2	759.7	1063.3	258.67	61.3
45	721	616	506	609	634.2	577.0	561	593.0	648.0	687.0	538.00	467.0
46	611	792	366	669	622.5	609.0	579	639.0	852.0	1035.0	426.00	183.0
47	503	892	320	675	625.1	629.0	606	652.0	938.0	1201.0	366.00	57.0
48	614	656	585	626	624.6	622.3	620.5	624.2	659.7	693.3	588.67	551.3

Thus, based on the data stimulated for the bandwidth consumption over a given period, the sheet depicted in the Table 1 representation indicates the scope of data. Based on the information depicted to 48th Time-Frame, the pivot points are calculated, as depicted in the spread sheet representation.

From the information, assessed in the spread sheet, the CPR conditions for the 47th Time-Frame, and 48th Time-Frame are depicted in the following graphical representation.

Table 2: Based on the data adapted for estimations

Pivots	**Time-Frame-47**	**Time-Frame-48**	**Moving Price**
Pivot Point	629.0	622.3	857
BC	606.0	620.5	
TC	652.0	624.2	
R1	938.0	659.7	
R2	1201.0	693.3	
S1	366.0	588.7	
S2	57.0	551.3	

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 10
DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

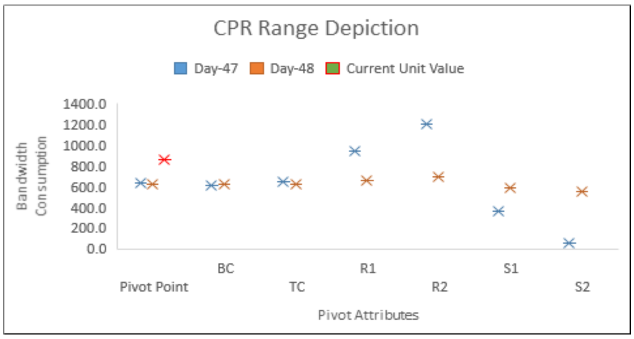


Figure 3: The conditions wherein CPR range estimations for the last two Time-Frames are estimated and plotted

Figure 3 representation refers to the conditions wherein CPR range estimations for the last two Time-Frames are estimated and plotted in the graphical representation. From the estimations in the above-mentioned table 2 of CPR calculations for the two Time-Frames, on Time-Frame47, the CPR range is in the range of 48 units, whereas on Time-Frame 48, the CPR range is 20 units.

The recorded values implicit that with a wider CPR range of Time-Frame 47, the scope of volatility is less, and on Time-Frame 48, with a CPR range of 20 points, the volatility can be higher. Thus, there is a need for more alert scrutiny of the performance of the current unit consumption metric.

As estimated with a random current moving price of 857, as depicted in figure 4, with a yellow background and red strike plotter, the movement is currently higher than R2, and it refers to substantial monitoring conditions.

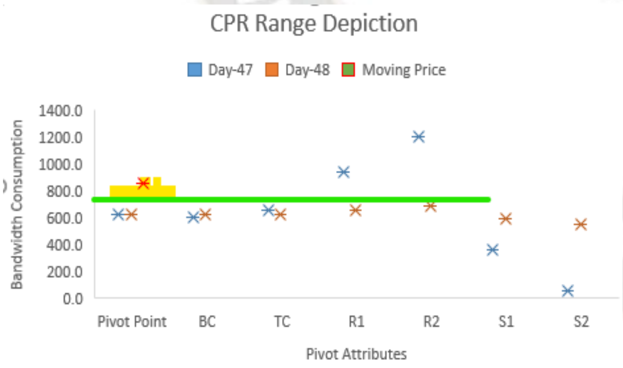


Figure 4: CPR- Range Detection

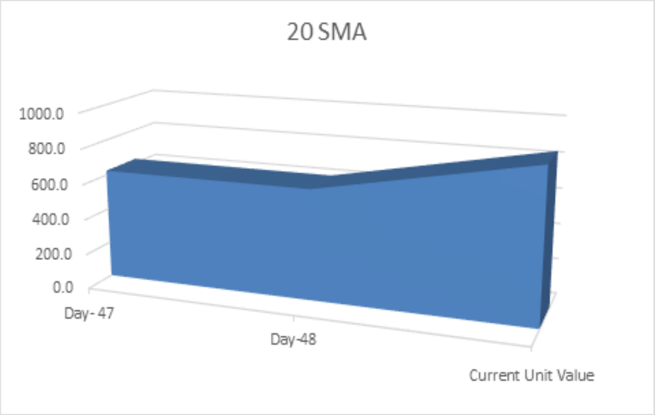


Figure 5: 20-Simple Moving Average (SMA) estimation

Based on the crossover of the resistance level R2, the model refers to the analysis of the 20-period SMA cross-analysis. The figure 5 representation depicted indicates that the current unit value M_{cpv} is higher than 20 SMA, thus indicating more significant volumes. Hence, the teams need to consider screening the t-value in the case of the Markov Chain models, and accordingly, if necessary, the actual malware detection models can be deployed for significant analysis and prevention act.

4.1 Cross-validation

Table 3: The table exhibiting average and deviation values

	CPH	SH-IDS	TLA-IDS
Sensitivity	0.945225±0.010672	0.907513±0.003841	0.786841±0.050589
Specificity	0.923125±0.002459	0.913751±0.002463	0.833425±0.010505
Accuracy	0.942551±0.012376	0.923421±0.006218	0.851451±0.008559
PPV	0.948652±0.021901	0.933775±0.018471	0.843153±0.022284
NPV	0.921325±0.007399	0.913625±0.006472	0.848451±0.007433

Table 3 is exhibiting the mean values and respective deviations of the cross-validation metrics obtained from the cross-validation performed on benchmark dataset UNSW-NB15 [23], [24], [25]. The values obtained from CPH are outperforming the other two contemporary models SH-IDS and TLA-IDS. The sensitivity and specificity of the proposed method CPH are exhibiting that both labels' detection accuracy is robust and far better than the contemporary models. The CPH model has approximately 7% more accuracy than SH-IDS and 12% more than TLA-IDS.

DOI: https://doi.org/10.17762/ijritcc.v10i10.5738
***Article Received**: 28 July 2022 **Revised**: 05 September 2022 **Accepted**: 12 September 2022*

4.1.1 Sensitivity

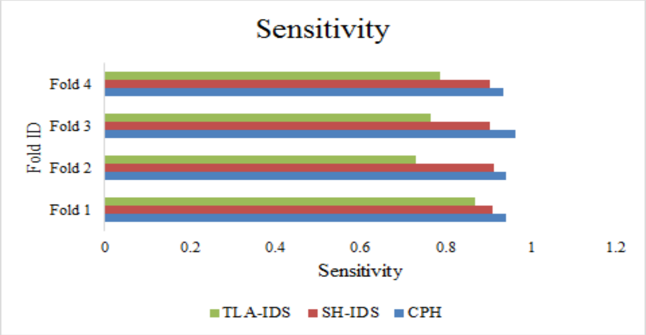


Figure 6: Sensitivity value noticed for proposed model CPH, and contemporary models SH-IDS and TLA-IDS

The metric sensitivity is also known to be Recall. The graph has been represented among 4 folds and sensitivity metrics. The performance of the proposed model CPH and the contemporary models TLA-IDS [15] and SH-IDS [16] have been measured with the metric sensitivity over the 4 folds as shown in figure 6. It has been perceived that sensitivity of the projected model CPH performs better when compared to TLA-IDS and SH-IDS.

4.1.2 Specificity

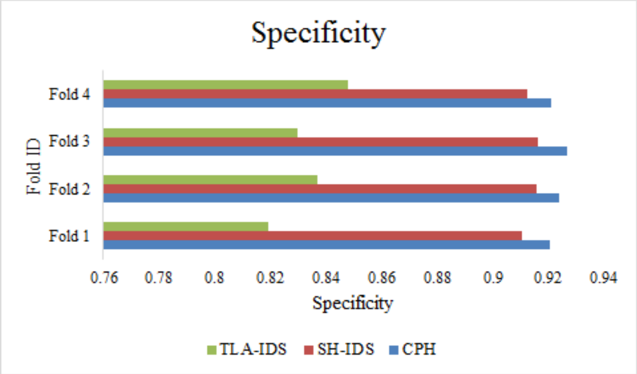


Figure 7: Specificity value noticed for proposed model CPH, and contemporary models SH-IDS and TLA-IDS

In figure 7, the metric specificity has been used for measuring the performance of the proposed approach CPH and contemporary models TLA-IDS and SH-IDS over the four folds as shown in figure 7. Specificity has been stated as a proportion of actual negatives, which predicted correctly. Based on the statistics portrayed in the above figure, it is noticed that specificity of the projected model CPH performs better and is added more advantage when compared to contemporary models TLA-IDS and SH-IDS.

4.1.3 Accuracy

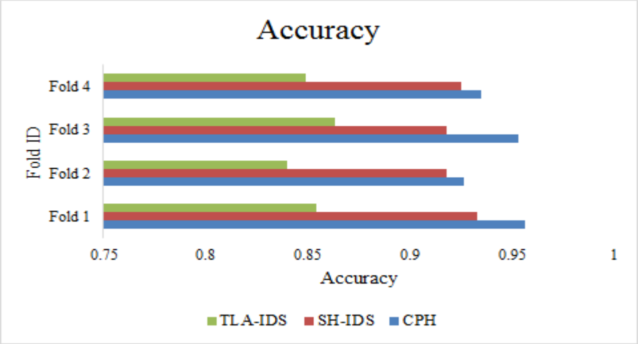


Figure 8: Accuracy value noticed for proposed model CPH, and contemporary models SH-IDS and TLA-IDS

One of the metrics used for measuring the classification models is accuracy. The graph has been represented among 4 folds and Accuracy metrics. The performance of the proposed model CPH and the contemporary models TLA-IDS [15] and SH-IDS [16] have been measured with the metric accuracy over the 4 folds as shown in figure 8. It has been perceived that accuracy of the projected model CPH performs better when compared to TLA-IDS and SH-IDS.

4.1.4 Positive predictive value (PPV)

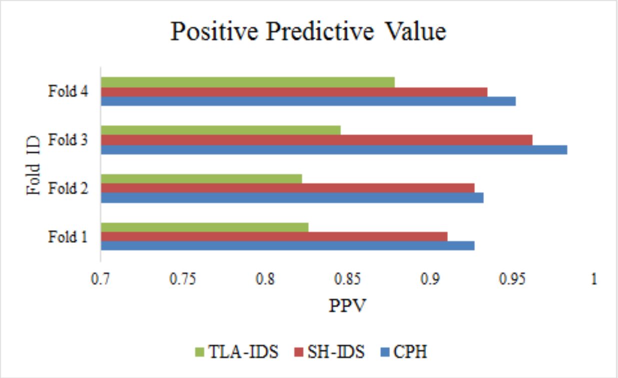


Figure 9: PPV (precision) value noticed for proposed model CPH, and contemporary models SH-IDS and TLA-IDS

In figure 9, the metric PPV has been used for measuring the performance of the proposed approach CPH and contemporary models TLA-IDS and SH-IDS over the four folds as shown in figure 9. Precision is also termed PPV. Based on the statistics portrayed in the above figure, it is noticed that PPV of the projected model CPH performs better and is added more advantage when compared to contemporary models TLA-IDS and SH-IDS.

4.1.5 Negative Predictive value (NPV)

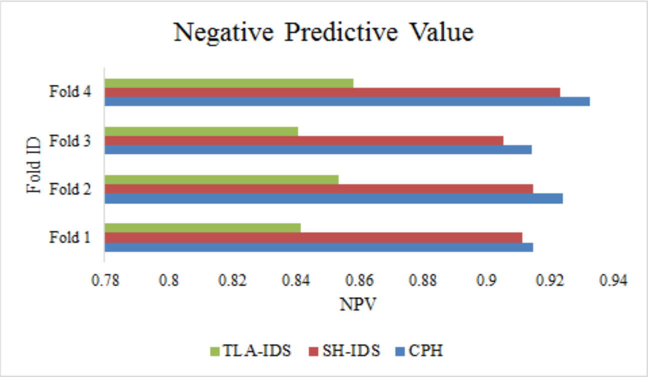


Figure 10: NPV noticed for proposed model CPH, and contemporary models SH-IDS and TLA-IDS

NPV is defined as the ratio of true negatives (TN) to the sum of TN and FN (false negatives). The graph has been represented among 4 folds and NPV metric. The performance of the proposed model CPH and the contemporary models TLA-IDS and SH-IDS have been measured with the metric NPV over the 4 folds as shown in figure 10. It has been perceived that NPV of the projected model CPH performs better when compared to TLA-IDS and SH-IDS.

5 Conclusion

Many contemporary solutions were imperative from the literature for managing the detection and prevention of botnet attacks. While certain significant solutions are resourceful to detect such attacks, the network and system administrations need to ensure there is minimal impact on the server performance due to background applications load consumption. Thus, in this manuscript, a four-layer detection surface model is proposed, which shall act as a screener referring to any early signs of botnet attacks.

The proposed solution is centric to CPR (central pivot range) at the center and supported by the estimations of support, resistance, moving average estimations, that drive the analysis for final filter screening in terms of Markov Chain set-up of time frame movement from one stage to the other. Based on the simulated data environment of bandwidth consumption analysis, the proposed model CPH is prototype analyzed, wherein the model has significant impact in terms of the lighter application being deployed for botnet tracing at IoT devices level. The performance of the CPH is scaled by comparing with the other contemporary models SH-IDS and TLA-IDS. In order to this, cross validation has performed using benchmark botnet dataset UNSW-NB15. The results obtained for standard metrics of cross validation performed on proposed model CPH and contemporary models SH-IDS, and TLA-IDS indicating that the proposed model CPH is outperforming the contemporary models with high sensitivity, specificity, and accuracy with minimal false alarming. The future research can focus to define access gateway level botnet detection strategies for IoT networks.

References

[1]. Dange, Smita, and Madhumita Chatterjee. "IoT Botnet: The Largest Threat to the IoT Network." Data Communication and Networks. Springer, Singapore, 2020. 137-157.

[2]. Acarali, Dilara, et al. "Modelling the spread of botnet malware in IoT-based wireless sensor networks." Security and Communication Networks 2019 (2019).

[3]. Lohachab, Ankur, and Bidhan Karambir. "Critical analysis of DDoS—An emerging security threat over IoT networks." Journal of Communications and Information Networks 3.3 (2018): 57-78.

[4]. Ceron, João Marcelo, et al. "Improving iot botnet investigation using an adaptive network layer." Sensors 19.3 (2019): 727.

[5]. Adi, Erwin, Zubair Baig, and Philip Hingston. "Stealthy Denial of Service (DoS) attack modelling and detection for HTTP/2 services." Journal of Network and Computer Applications 91 (2017): 1-13.

[6]. Abaid, Zainab, et al. "All Infections are Not Created Equal: Time-Sensitive Prediction of Malware Generated Network Attacks." arXiv preprint arXiv:2102.01944 (2021), pp-14.

[7]. Rhee, Junghwan, et al. "Defeating dynamic data kernel rootkit attacks via vmm-based guest-transparent monitoring." 2009 international conference on availability, reliability and security. IEEE, Fukuoka, Japan, 2009, pp. 74-81.

[8]. Soe, Yan Naung, et al. "Machine Learning-Based IoT-Botnet Attack Detection with Sequential Architecture." Sensors 20.16 (2020): 4372.

[9]. Almutairi, Suzan, et al. "Hybrid Botnet Detection Based on Host and Network Analysis." Journal of Computer Networks and Communications 2020 (2020).

[10]. Gudni Johannesson, & Nazzal Salem. (2022). Design Structure of Compound Semiconductor Devices and Its Applications. Acta Energetica, (02), 28–35. Retrieved from http://actaenergetica.org/index.php/journal/article/view/466

[11]. Ersson, Joakim, and Esmiralda Moradian. "Botnet Detection with Event-Driven Analysis." Procedia Computer Science 22 (2013): 662-671.

[12]. Alqahtani, Mnahi, Hassan Mathkour, and Mohamed Maher Ben Ismail. "IoT Botnet Attack Detection Based on Optimized Extreme Gradient Boosting and Feature Selection." Sensors 20.21 (2020): 6336.

[13]. Htwe, Chaw Su, Yee Mon Thant, and Mie Mie Su Thwin. "Botnets Attack Detection Using Machine Learning Approach for IoT Environment." Journal of Physics: Conference Series. Vol. 1646. No. 1. IOP Publishing, 2020.

[14]. Kaur, Navdeep, and Maninder Singh. "Botnet and botnet detection techniques in cyber realm." 2016 International Conference on Inventive Computation Technologies (ICICT). Vol. 3. IEEE, 2016.

[15]. https://www.investopedia.com/trading/using-pivot-points-for-predictions/

[16]. Deng, Lianbing, et al. "Mobile network intrusion detection for IoT system based on transfer learning algorithm." Cluster Computing (2018): 1-16.

[17]. Babu, M. Jagadeesh, and A. Raji Reddy. "SH-IDS: Specification Heuristics Based Intrusion Detection System for IoT Networks." Wireless Personal Communications (2020): 1-23.

[18]. Rolf Bracke, & Nouby M. Ghazaly. (2022). An Exploratory Study of Sharing Research Energy Resource Data and Intellectual Property Law in Electrical Patients. Acta Energetica, (01), 01–07. Retrieved from http://actaenergetica.org/index.php/journal/article/view/459

[19]. https://tradingtuitions.com/all-you-wanted-to-know-about-central-pivot-range-cpr-indicator/

[20]. Kemeny, John G., and J. Laurie Snell. Markov chains. Vol. 6. Springer-Verlag, New York, 1976.

[21]. M. N. Prasad* et al., "Reciprocal Repository for Decisive Data Access in Disruption Tolerant Networks," International Journal of Innovative Technology and Exploring Engineering, 2019, 9(1), pp. 4430–4434

[22]. Reddy, K. Uday Kumar, S. Shabbiha, and M. Rudra Kumar. "Design of high-security smart health care monitoring system using IoT." Int. J 8 (2020).

[23]. Thanigaivelan NK, Nigussie E, Kanth RK, Virtanen S, Isoaho J. Distributed internal anomaly detection system for Internet-of-Things. In Consumer Communications & Networking Conference (CCNC), 2016 13th IEEE Annual 2016 Jan 9 (pp. 319-320). IEEE.

[24]. Rudra Kumar, M., Rashmi Pathak, and Vinit Kumar Gunjan. "Machine Learning-Based Project Resource Allocation Fitment Analysis System (ML-PRAFS)." Computational Intelligence in Machine Learning. Springer, Singapore, 2022. 1-14

[25]. Gunjan, Vinit Kumar, and Madapuri Rudra Kumar. "Predictive Analytics for OSA Detection Using Non-Conventional Metrics." International Journal of Knowledge-Based Organizations (IJKBO) 10.4 (2020): 13-23

[26]. The-UNSW-NB15-dataset, March 2018. [Online]. Available: https: //www.unsw.adfa.edu.au/australian-centre-for-cybersecurity/ cybersecurity/ADFA-NB15-Datasets/.

[27]. N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in Military Communications and Information Systems Conference (MilCIS), 2015. IEEE, 2015, pp. 1–6.