Multicultural Education

Volume 7, Issue 11, 2021

Using Machine Learning, An Intrusion Detection and Prevention System for Malicious Crawler Detection in e-Learning Systems

Dinesh Mavaluru

Article Info

Article History

Received:
April 22, 2021

Accepted:
November 24, 2021

Keywords :
Intrusion Detection,
Malicious Crawlers,
Bayesian Network,
Support Vector Machine,
e-learning systems

DOI:
10.5281/zenodo.5725307

Abstract

In the modern era, e-Learning creates various remotely managed platforms for users to learn online. However, the security risk to the education sector is also elevated, as evidenced by the increasing number of reported attacks on these e-learning systems. The study says that the warning of the crawler's cyber-attacks is increasing. Malicious Crawlers can crawl webpages, crack passwords, and steal the user's personal information. Furthermore, in a dynamic environment, intrusion detection generates more false positives. This research work developed an efficient approach for malicious crawler detection and correlated security alerts to compare various machine learning strategies such as Bayesian Network, Support Vector Machine, and Decision Tree. Then this optimized model was established by evaluating the fitted features and accuracy. This study tested and validated the efficacy of the proposed methodology for detecting actual attacks while producing low levels of false positives.

Introduction

The most difficult challenge in the e-learning sector is protecting user data from malicious crackers [1]. The user portal in e-learning Infrastructure is involved in various sensitive functions, student transactions, and interactions with educators. One of the essential security features in a user portal is dealing with password crackers and data thieves. Malicious Crawlers are becoming more prevalent by the day. The demand for retrieving data from the e-learning network has resulted in various types of Crawlers [2 - 5]. It is critical to detect the crawlers' design goals and routing patterns to identify them. [6] It facilitated action based on their routing style and requested data type. Crawlers are designed with various goals in mind, such as spying and stealing data, trading information. [7]Detecting the crawlers' functional goals remains difficult. Crawlers launch Denial of Service (DoS) attacks and analyze traffic [8]. Furthermore, with the introduction of new features such as wireless, peer-to-peer networks, and memory keys, e-learning infrastructure is becoming increasingly complex. The Internet standardized the complex attacks injected by downloads. As a result, the number of false alerts has increased [9 - 11]. Because user information is compassionate in the e-Learning infrastructure, it is critical to always protect user data and in all places.

This research work contributes to the identification of malicious crawlers by inspecting crawler behavioral aspects. These crawler behavioral characteristics aided in identifying known, ethical, unethical, malicious, and unknown crawlers. It aids in the reduction of Password Cracking, Data Reaping, and DoS Attacks.Various machine learning techniques are used to identify the various crawlers perfectly. Also, to deal with the excess false-positive rates, we proposed two significant concepts in machine learning solutions. The first concept is event fusion to meta events. The second is the categorization of meta-events.

The work in [12] started the research to detect crawlers using a model of navigational pattern that relies on decision trees. C4.5's competence was reduced because of the use of various continuous attributes. Furthermore, it generates complicated rules.[13]enhanced classification techniques with highly fitted attributes to present a variety of attributes based on structure and web content. According to Lee et al., feature matching usage, structure, and web content may play a role in Malicious Crawler detection.[14] conducted a ranking analysis after introducing two groups of attributes, namely resources and workload. [15 -16] examined the various behavioral characteristics of search engines in the log files of five academics and attempted to identify the crawlers by analyzing the attributes. It was discovered that the current attributes varied depending on the type of websites and crawlers.

As a result, [17] developed the correlation hierarchy and its attack. The sensor detects security alerts and spreads them across all networks; this information is then compiled into meta-alerts using one of three approaches: Synthetic threads (a) (b) A security incident occurred, and (c) reports of related attacks were received. Ning et al. addressed the detection of multi-stage attacks by mentioning the attack's source, need, and effect as first-order logic predicates. Hyper-alerts are created by combining various levels of single attacks. As a result, it aids in reducing the number of security alerts.

The work in [18] used Bayesian Multiple Hypotheses Tracking in the fusing sensor process output to achieve awareness and respond quickly from the security side. The work in [19 - 20] addressed the correlation of events in the Bayesian Network fragment, which reduces the number of alerts examined by the security side. The work in [21] compared the productivity of novel attributes and various types of feature extraction classification. The work in [22] presented a model for spambot detection that consists of three approaches: extractor, tracker, and classifier. There are other functional phenomena at work in detecting aspects and crawler behavior. The work in [23 - 24] investigated the crawler's characteristics for detecting the requested file series The work in [25] worked on a scholarly information model for detecting Malicious Crawlers.Crawler behavioral features are analyzed in this research work for malicious crawler detection, and a new methodology is proposed to maximize security in the E-Learning user portal. Furthermore, it correlates security events using machine learning approaches. However, the SVM classifier and the Bayesian classifier can be used to examine the accuracy level.

1. Model Suggested

This study helps to improve the security of user portals and websites in the e-Learning infrastructure. Log files contain information about HTTP webserver traffic. The standard log file input is compared to the web server's response and the user's request. The requests in the log file are in the form of independent events and are placed at specific time intervals. These files must be processed to extract the user agent's behavioral characteristics. Figure 1 depicts the methods for detecting Malicious Crawlers. Furthermore, for effective security management in e-Learning infrastructure, the correlation of security events is critical. This proposed work analyzed all alerts more profoundly and holistically to reduce false positives or false alerts in E-Learning Infrastructure.

In this proposed work, we fused events from various sources and normalized them to process them consistently. These normalized events are categorized and organized into meta events and groups. When compared to single events, meta events contained a more comprehensive description of likely attack cases and established a more sophisticated expression of casual attacks than isolated alerts [26]. Fusing everyday events into meta events helps to improve system performance. Meta events have deviated from previously learned attack strategies. In this case, Machine Learning can assist in automating these practices.

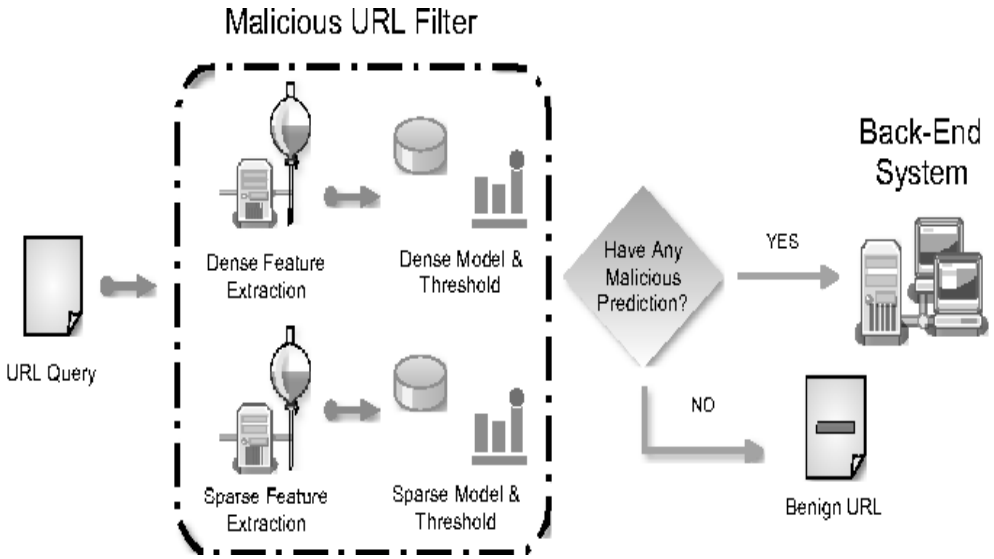

Figure 1. Malicious URL filtering

1.1 Pre-processing Log Files and Session Mining

The request for HTTP demands at the specific stretch is considered a meeting. The information in each period is referenced in the all-encompassing development of the client in entries and sites. This meeting follows

the navigational credits of every client that ought to be demanded the information from log records. The best HTTP demand assortment relied upon the client break meeting and IP address [27]. The current methodology is essential, yet the principal issue is that the current methodology is to consider the client's break meetings [26]. Most of the current exploration referenced that the break may be considered as 30 minutes. However, the log records perception shows that the break should be referenced while the single meeting is partitioned into different meetings.We proposed a program that dynamically fragments sessions based on user-agent and IP addresses to address this issue. As a result, the previous session referrer values are used if the period is near 30 minutes or more.

1.2 Featured Session Extractions

We extricated the appropriate ascribes out of the made meeting. Every single Session in log documents contains the requester's IP address, and the utilized convention draws near, time solicitation and date demand, status code, mentioned URL page, the size of referrer page, and mentioned text. Because of the differentiation conduct of clients and marine parts of crawlers, we can identify the crawlers even more viably from the highlights extricated from log documents. To check the removed highlight's effectiveness, we did the relationship investigation.

1.3 Label the Sessions

It is critical to label the Session for the classification algorithm. The session labeling algorithm is referred to as algorithm 1. Except for the following attributes, all sessions are assumed to be linked to the typical user.

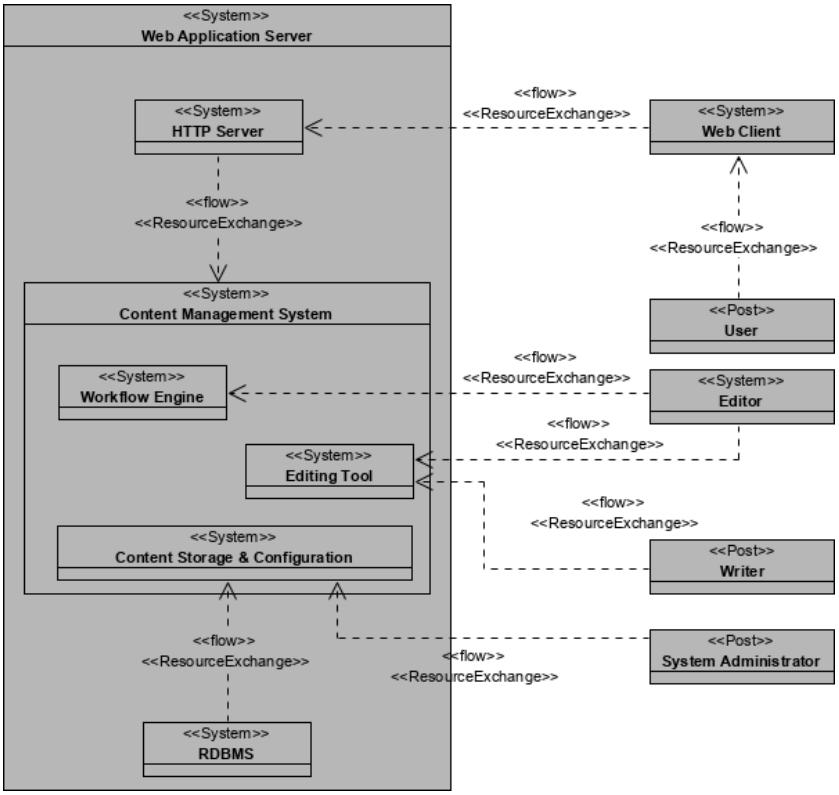

Figure 2. Systems Interface Description

1.4 Layers in a hierarchy

As a result, our proposed method has three layers: collection, normalization, classification, and fusion. The three layers are depicted in Figure 2 at each stage, and the abstraction level is provided. We processed the raw data and converted it into a consistent Intrusion Detection system. [31] Message Exchange Format We took the extended records and clustered them into the normalized meta-alerts during the fusion stage. Finally, during the classification stage, we classified all the meta-alerts as alarms or attacks.

Firewalls, IDs, SNMP traps are the occasion sources that are connected to Collection and Normalization. Assortment components are arranged into two areas as dynamic components and aloof components. The detached components' work is to assemble all alarms by investigating component states, occasions, and proof. Crafted by dynamic components is to speak with the overseen alarms for displaying the

cautions. In a portion of the cases like the SNMP trap, active alarm happens at each condition of the component or approaching occasion. Now and again, assortment components channel the approaching component and build some cordial alarms. Working System (OS) calls are one of the common occurrences for this. The log framework's subset compares to the dormant assaults. Standardization guarantees that the active alarms from the financial entryway are signified in a standardized manner.

We used enhancement of the linearized IDMEF standard in this proposed methodology. To outperform the XML bus, we used linearization methodology in our system [32]. We now used the most straightforward method for alert identification, which should aid in the classification and fusion processes. This methodology detects each alert by assigning a single value to each alert representing the type of alert, and because the taxonomy naming space must be determined, it is used in the classification layer of a few algorithms. Real-world alerts and natural language expression methodology were also used to achieve finitude and expressiveness.Alert detections are derived from several sources, including firewall incidents, Device IDS calls, Windows Authentication, web server log access, and so on

Objects (Obj) are denoted as a multi-level hierarchy in this context. The Object's topmost level entity in the framework is a network element, a series of protocols, a portal user, file, and protection. Each variable is divided or broken into a more significant number of granular schemes. If there is a need, the hierarchy could have been expanded to include other managed components.Activity is like this proposed approach, in which it is associated with the article. Thus, a caution about the framework rest would specify like os: rest. Activity is additionally meant as a staggered progression as Objects. The lower-level chain of command qualifies upper levels. Model: a login endeavor disappointment as the framework organization because of wrong OTP or manual human test, would indicate as the user.admin.login.fail.captcha. The accompanying portrayals develop activities.

Subj. represents the subject. Adv. Represents Adverbial, and Obj represents Object. Action word transitiveness is the significant contrast between the two cases. Other than that, this vulnerability will not have a commonsense impact on scientific classification. To address this vulnerability is consistently troublesome. A portion of the e-Learning cautions does not mean the activity against objects; however, it may address the states of the articles. These conditions are referenced as Subject+Copulative_Verb+Complement. For instance, the scientific categorization would be referenced as protocol if the TCP port is blocked off.TCP.port:inaccessible. Some classes of cautions referenced the doubts by their proof of movement.

In the vulnerability of inborn, our proposed set of scientific classification is doubts from conditions and activities. It varies from past situations in which it referenced not current realities but rather assumptions associated with current realities. The indirect access endeavor in our framework is likewise an illustration of vulnerability referenced as os: malware.backdoor.attempt. Here we referenced the articles, conditions, activities, and doubts in the staggering chain of command, which licenses the relationship to handle the granularity level that is more agreeable for each situation. The result of this technique can be ready to assign a solitary limited depiction for security alert. This portrayal ought to be gotten from genuine cautions and comprised of the plans of everyday language and pecking order and limited.Else, it deposits clear. All potential watchwords of the touch cluster set are associated with the noxious assaults. Potential assaults that appeared by meta-alarms happen in the subset of watchwords. The exhibit bit gives help to inside item philosophies which demonstrate the likeness between two instances of grouping layer.

Proposed approach produces more promising results in terms of Data to Information Ratio (DIR) and provides improved support to the classification layer, as will be shown later. More advanced functions, such as portal users, sessions, files, processes, are used for fusion. However, one drawback of this algorithm is that it does not produce promising results when alarms are merged for multi-stage or hybrid attacks. It is up to us to overcome these restrictions in our future studies.

2. Classification

The essential objective of demonstrating the crawler's characteristic example is to gauge the various classes of obscure crawlers through fantastic preparation. Subsequently, it is fundamental to pick a suitable model for information characterization. The most encouraging technique for managed learning is the Support Vector Machine (SVM) [33]. We can get precise and effective outcomes in Data mining and Pattern Recognition through SVM. SVM is additionally entirely appropriate for distinguishing the crawlers with a more substantial number of traits.

In our examination, information preprocessing and meeting details are not liberated from blunder stages. Hence, we need to choose an ideal classifier to decrease the current blunders. Furthermore, piecework choice in the characterization of SVM is essential. In our proposed approach for distinguishing malevolent crawlers in the financial gateway, we utilized RBF, Polynomial, and direct piece work.

3. Data Training

In this suggested technique, many types of datasets were investigated. We should check numerous types of log documents for a more prominent quantity of malicious crawlers' locations; consequently, the information was derived from diverse records and variable sources. The datasets used in this study should be tweaked to improve the overall performance of the web mining program.It implies that information associated with several classes should be comparable; otherwise, if the information is deemed unbalanced, the test records may be given the name of the dominating component class.

One of the most important solutions to this problem is to provide some information. Oversampling occurs when the need is enlarged. It is under-examined if the requirement is reduced. In any event, these data are nearly identical to one another. The oversampling model is used to aid all information in this investigation piece.However, test return in preparing and marking will play an important role in outcome determination because arrangement and assessment are gotten from the influence of modifying data.

Datasets	Total Session count	Number of User's sessions	Number of Malicious Crawler's sessions	View of Mean Page	Total Hit	Total bandwidth value occupied
Log File 1	14510	12142	6104	6.85	519886	12.47 GB
Log File 2	3500	11430	1224	7.52	11325	853.2 MB
Log File 3	4320	11324	4758	2.54	13123	913.2 MB
Log File 4	4630	17534	6789	5.6	15231	413.65 MB
Log File 5	5210	13554	1452	4.56	32313	732.53 B

4. Results

4.1 Error Rate and Accuracy

The error rate and classification accuracy were used to evaluate crawler detecting skills. As shown below, crawlers and humans are employed in groups to discover crawlers:

$$Error\ Rate\ = \frac{Number\ of\ uncorrected\ Predicted\ Session}{Total\ Number\ of\ Predictions}$$

$$Accuracy\ A = \frac{Number\ of\ Correctly\ Predicted\ Sessions}{Total\ Number\ of\ Predictions}$$

4.2 Precision, Recall, and F1- measure

Another measure that applies to both the balanced and unbalanced datasets is Precision, Recall, and F1-measure, which is mentioned below:

$$Precision\ P = \frac{Number\ of\ Crawlers\ Correctly\ Classified\ Sessions}{Number\ of\ Predicted\ Crawlers\ Sessions}$$

$$Recall\ R = \frac{Number\ of\ Crawlers\ Correctly\ Classified\ Sessions}{Number\ of\ Actual\ Crawler\ Sessions}$$

$$F1 - measure = \frac{2PR}{P + R}$$

4.3 Evaluation of ImplementationandExecution

We utilized the SoTM (Scan of The Month) data sets from the Honey Network Project [32] to test our technique. This information base is openly accessible and, it shows the simple assaults done by wafers. Figure 3 and Figure 4 represent a clear perspective on model execution. For Data Classification and Analysis, we utilized 80% of the datasets for preparing the datasets, and 20% was utilized for testing.

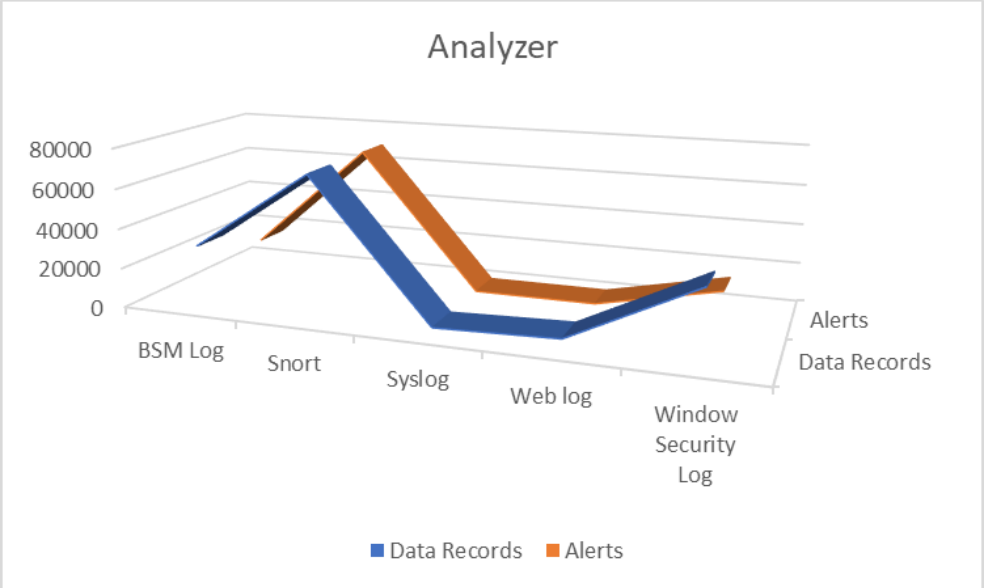

Figure 3. Systems Analyzer for Alerts and Data Records

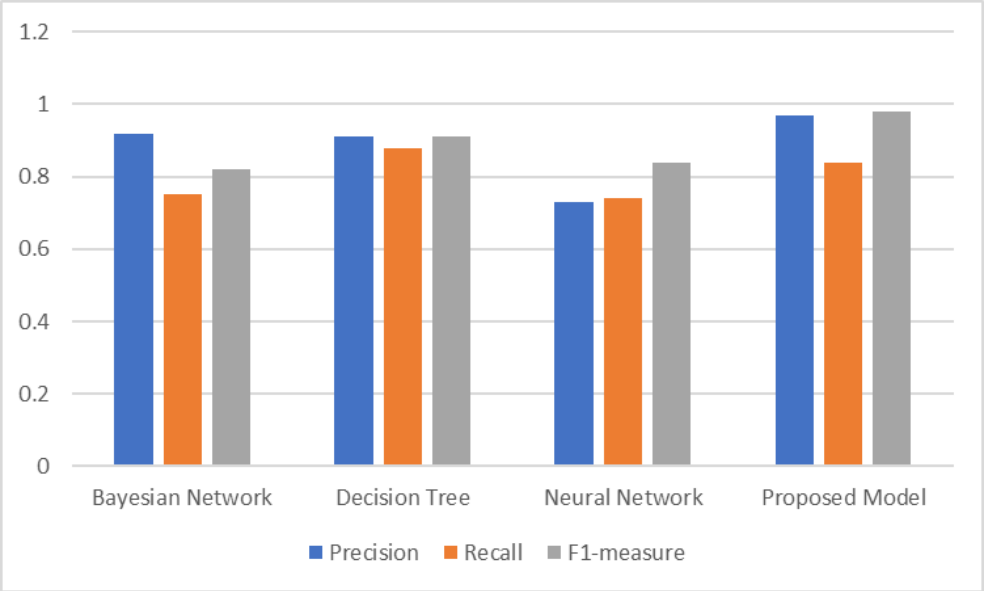

Figure 4. Precision, Recall & F1 Measure with New Attributes

The model of revamping and resampling was repeated in any event multiple times, and the last dataset's result was recovered from the connected normal. The meeting needs at least three solicitations for meeting marking and removing it ascribes. Subsequently, the more limited length meetings were not being taken. More extended length meeting boosts the exactness of namingcorrelation between the exactness level of the combined dataset without new highlights and with new highlights. The Proposed strategy comprises the new credits of HIT, Request Reappearance Percentage, and Unique Pages. Spotting crawlers from various mixes of information extricated from different sites are exceptionally troublesome. Be that as it may, the proposed model identifies a high percent of pernicious crawlers.

Conclusion

This research was expanded in numerous ways. Initially, the findings are presented for a variety of log file kinds. The unique locations from separate log data are therefore selected for the generation of appropriate datasets. Session Affinity, Attribute Mining, Session Marking, and Categorization were all held. Furthermore, this technique separated meta-alerts into higher-level meta-alerts for fusing multi-stage attacks and various kinds of threats.For much more consistent categorization, this strategy employed progressive clustering approaches and assessed the probability of existing topologies in SVM classifiers. It also enhanced taxonomy in several disciplines. A combination of approaches, such as a genetic algorithm combined with a more popular classifier, is suggested for excellent feature collecting in future investigations.

References

Besanko and A. V. Thakor, “E-Learning deregulation: Allocational consequences of relaxing entry barriers,” J. E-Learning. Financ., vol. 16, no. 5, pp. 909–932, 1992, doi: 10.1016/0378-4266(92)90032-U.

Blumhardt, W. Gaul, and L. Schmidt-Thieme, “Web robot detection - Preprocessing web logfiles for robot detection,” Stud. Classif. Data Anal. Knowl. Organ., vol. 0, no. 211289, pp. 113–124, 2005, DOI: 10.1007/3-540-27373-5_14.

D. J. Burroughs and L. F. Wilson, G. V. Cybenko, "Analysis of distributed intrusion detection systems using bayesian methods," In Proceedings of IEEE International Performance Computing and Communication Conference., 2002, pp 329–334.

D. Stevanovic, N. Vlajic, and A. An “Detection of malicious and non-malicious website visitors using unsupervised neural network learning,” Appl. Soft Comput. J., vol. 13, no. 1, pp. 698–708, Jan. 2013, doi: 10.1016/j.asoc.2012.08.028.

F. T. Ammari, J. Lu, and M. Aburrous, “Intelligent E-Learning XML Encryption Using Effective Fuzzy Logic,” in Emerging Trends in ICT Security, Elsevier Inc., 2013, pp. 591–617.

H. Kayacik, … A. Z.-H.-P. of the, and undefined 2005, “Selecting features for intrusion detection: A feature relevance analysis on KDD 99 intrusion detection datasets,” pdfs.semanticscholar.org, Accessed: Jun. 17, 2020. [Online]. Available: https://pdfs.semanticscholar.org/1d6e/a73b6e08ed9913d3aad924f7d7ced4477589.pdf.

J. Lee, S. Cha, D. Lee, and H. Lee, “Classification of web robots: An empirical study based on over one billion requests,” Comput. Secure., vol. 28, no. 8, pp. 795–802, 2009, DOI: 10.1016/j.cose.2009.05.004.

J. W. Greene, "Web robot detection in scholarly Open Access institutional repositories," Libr. Hi-Tech, vol. 34, no. 3, pp. 500–520, 2016, DOI: 10.1108/LHT-04-2016-0048.

K. Bajaj, A. A. Chitkara, and H. Pradesh, “Improving the Intrusion Detection using Discriminative Machine Learning Approach and Improve the Time Complexity by Data Mining Feature Selection Methods,” 2013. Accessed: Jun. 17, 2020. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.481.8435&rep=rep1&type=pdf.

K. Stroeh, E. R. M. Madeira, and S. K. Goldenstein, “An approach to the correlation of security events based on machine learning techniques,” J. Internet Serv. Appl., vol. 4, no. 1, pp. 1–16, 2013, DOI: 10.1186/1869-0238-4-7.

M. Bahrololum, … E. S.-2009 F. I., and undefined 2009, “Machine learning techniques for feature reduction in intrusion detection systems: A comparison,” ieeexplore.ieee.org, Accessed: Jun. 17, 2020. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5369962/.

M. T. Qassrawi and H. Zhang, “Detecting Malicious Web Servers with Honeyclients,” pdfs.semanticscholar.org, 2011, DOI: 10.4304/jnw.6.1.145-152.

M. Wu and Y. Moon, “Alert Correlation for Cyber-Manufacturing Intrusion Detection,” Procedia Manuf., vol. 34, pp. 820–831, 2019, DOI: 10.1016/j.promfg.2019.06.197.

Mosh Chuk, T. Bragin, S. D. Gribble, and H. M. Levy, “A Crawler-based Study of Spyware on the Web.” Accessed: Jun. 18, 2020. [Online]. Available: http://courses.cs.washington.edu/courses/cse454/15wi/papers/spycrawler.pdf.

N. Hosseini, F. Fakhar, B. Kiani, and S. Eslami, “Enhancing the security of patients’ portals and websites by detecting malicious web crawlers using machine learning techniques,” Int. J. Med. Inform., vol. 132, no. March 2019, DOI: 10.1016/j.ijmedinf.2019.103976.

P. Hayati, V. Potdar, K. Chai, and A. Talevski, “Web spambot detection based on web navigation behavior,” Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA, pp. 797–803, 2010, DOI: 10.1109/AINA.2010.92.

P. N. Tan and V. Kumar, “Discovery of web robot sessions based on their navigational patterns,” Data Min. Knowl. Discov., vol. 6, no. 1, pp. 9–35, 2002, DOI: 10.1023/A:1013228602957.

S. H. Kang and K. J. Kim, “A feature selection approach to find optimal feature subsets for the network intrusion detection system,” Cluster Comput., vol. 19, no. 1, pp. 325–333, Mar. 2016, DOI: 10.1007/s10586-015-0527-8.

S. Kwon, M. Oh, D. Kim, J. Lee, Y.-G. Kim, and S. Cha, “Web Robot Detection based on Monotonous Behavior,” Proc. Inf. Sci. Ind. Appl., pp. 43–48, 2012, [Online]. Available: www.microsoft.com

S. Kwon, Y. G. Kim, and S. Cha, “Web robot detection based on pattern-matching technique,” J. Inf. Sci., vol. 38, no. 2, pp. 118–126, 2012, DOI: 10.1177/0165551511435969.

Sabata, "Evidence aggregation in hierarchical evidential reasoning. In UAI Applications Workshop", Uncertainty in AI, 2005.

Sabata, C. Ornes, "Multisource evidence fusion for cyber-situation assessment". In: Proc. SPIE, 2006, Vol. 6242.

Stassopoulou and M. D. Dikaiakos, “A Probabilistic Reasoning Approach for,” pp. 265–272, 2007.

Stassopoulou and M. D. Dikaiakos, “Web robot detection: A probabilistic reasoning approach,” Comput. Networks, vol. 53, no. 3, pp. 265–278, 2009, DOI: 10.1016/j.comnet.2008.09.021.

Stevanovic, A. An, and N. Vlajic, “Feature evaluation for web crawler detection with data mining techniques,” Expert Syst. Appl., vol. 39, no. 10, pp. 8707–8717, 2012, DOI: 10.1016/j.eswa.2012.01.210.

T. Pietraszek and A. Tanner, “Data Mining and Machine Learning-Towards Reducing False Positives in Intrusion Detection *.” Accessed: Jun. 17, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1363412705000361.

T. Pietraszek, “Using adaptive alert classification to reduce false positives in intrusion detection,” Lect. Notes Comput. Sci. (including Subsea. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 3224, pp. 102–124, 2004, DOI: 10.1007/978-3-540-30143-1_6.

Valdes and K. Skinner, “Probabilistic alert correlation,” Lect. Notes Comput. Sci. (including Subsea. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 2212, pp. 54–68, 2015, DOI: 10.1007/3-540-45474-8_4.

Y. M.-P. applications of intelligent systems and undefined 2011, “Adaptive false alarm filter using machine learning in intrusion detection,” Springer, Accessed: Jun. 17, 2020. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-25658-5_68.

Author Information

Dinesh Mavaluru
Department of Information Technology, College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia