International IT Journal of Research (IITJR)
Volume 2, Issue 4, October- December, 2024
Available online at: https://itjournal.org/index.php

Zero-Day Malware Detection

Saurabh Kansal

Independent Researcher, USA

ABSTRACT

The phrase "zero-day malicious software" (malware) describes a recently identified or unidentified software vulnerability. Enhancing detection for similar zero-day malware by effective learning to plausibly produced data is the main goal of this work. Hardware-supported Malware Detection (HMD), which uses Machine Learning (ML) techniques applied to Hardware Performance Counter (HPC) data, has proven effective in detecting malware at the micro-architecture level of processors in order to overcome the high complexity of traditional software-based detection techniques. In this study, we investigate the appropriateness of many common machine learning classifiers for zero-day malware detection on novel data streams in the actual operation of Internet of Things devices. We show that these approaches are unable to provide a high detection rate for unknown malware signatures. We begin our study by reviewing current ML-based HMDs that use information from built-in HPC registers. We next investigate the appropriateness of several common machine learning classifiers for zero-day malware detection and show that they are unable to identify unknown malware signatures with a high detection rate.Last but not least, we suggest an ensemble learning-based method to improve the performance of the conventional malware detectors in order to overcome the difficulty of run-time zero-day malware detection, even if it only uses a few micro architectural elements that are recorded at run-time by current HPCs. The experimental results show that our suggested method, which uses only the top 4 micro architectural features to detect zero-day malware, achieves 92% F-measure and 95% TPR with only 2% false positive rate when Ada-Boost ensemble learning is applied to Random Forrest classifier as a regular classifier.

Keywords: -Zero-Day, Malicious Software, Random Forrest, HMDs, Iot, Hardware Performance Counter (HPC), Detecting Unknown, Experimental Results, Machine Learning (ML) Techniques, Ada-Boost.

INTRODUCTION

With more than 2.5 billion Android devices in use, including wearables, smartphones, and in-car applications, Google's Android operating system is extensively utilized [1]. This makes it easier for malware writers to launch attacks, which is why efficient methods for detecting Android malware are necessary [1, 2]. When malicious applications manage to hack a device, they may create serious problems, such as identity theft, financial loss, and the exfiltration of private information [2, 3]. Android-powered cars with in-car systems or automated driving are also vulnerable to assaults that might jeopardize the safety of both drivers and pedestrians.

The marketplace has a huge number of applications [2, 3], making it impossible to manually verify each one's validity. Earlier studies used machine learning (ML) methods to try to automate detection on a large scale [2, 4]. However, to manually create input characteristics for such detectors, a thorough understanding of Android malware is often needed [2, 6]. This may make them more susceptible to zero-day attacks until the new malware can be identified and reverse-engineered, after which detection features are improved [2, 8].

Our goal in this study is to disentangle these domain insights from the learning process and input data. Consequently, our approach is appealing for practical implementation and upkeep, especially when considering zero-day vulnerabilities [5, 6]. Because there is no known defense against the malicious action in issue, zero-day malware is the most hazardous; hence, the name "zero-day" [5] refers to malware that is first found at the moment of the assault. This leads into situations where a certain malware family has never been seen previously in the Android domain [8, 9].

Historically, several software-level security mechanisms have been used to guarantee data integrity, with the underlying hardware being presumed to be safe [19]. With a rising number of hardware assaults being revealed, this assumption is no longer valid. The system is mostly burdened with computational overheads and complexity due to the inefficiency of traditional software-based malware detection approaches. These detection techniques are also reliant on the examination of the apps' static signatures, which hinders their ability to identify hidden assaults during runtime [10, 11]. By using low-level micro architectural aspects of running apps on the target system, Hardware-Supported Malware Detection (HMD) has developed as a solution to the computational overheads and performance issues of standard malware detection approaches. These characteristics are gathered via Hardware Performance Counters (HPCs) registers [12], which are special-purpose registers built into contemporary microprocessors to record hardware-oriented events of profiled programs that remain on the processor architecture below. Hardware-assisted malware detection techniques

have shown that common machine learning (ML) algorithms used on HPC data are appropriate for identifying harmful application patterns [12, 13].

Using hardware characteristics that have been overlooked in previous HMD investigations, we have tackled the problem of identifying zero-day malware patterns at run-time in this study. In particular, our thorough analysis of various malware types and machine learning algorithms for HPC-based malware detection shows that standard machine learning classifiers, which have been widely used in previous works, are unable to detect zero-day (unknown) malware with a high detection rate [14]. Standard ML classifiers used for zero-day malware detection clearly perform worse, according to our findings [13]. We begin our effort by analysing current machine learning (ML)-based malware detection techniques that use information from built-in HPC registers. We next thoroughly analyse the applicability of many common machine learning classifiers for zero-day malware detection and show that they are unable to identify unknown malware signatures with a high detection rate [15].

Last but not least, we provide an ensemble learning-based method to improve the performance of the conventional malware detectors in order to tackle the problem of run-time zero-day malware detection [15], even if it only uses a few micro architectural elements that are recorded at run-time by current HPCs [14].

RELATED WORK

Hardware Performance Counters for Security Analysis

The computers and networks of today are much more complicated than they were a few decades ago. Out-of-order execution units, concurrent multithreading, and hierarchical cache subsystems and processor pipelines all significantly affect computing system performance. A key component of contemporary microprocessors (such as Intel, AMD, ARM, and [17]), the performance monitoring module [14] is often accessible via programmable hardware performance counter registers. Modern microprocessors are equipped with specialized registers called HPCs that are intended to track and record various hardware-related events. HPCs are restricted in the number of events that may be counted simultaneously due to a lack of physically costly HPC registers on the processor chip [18, 19].

ML for Hardware-Supported Malware Detection

Table 1 provides an overview of current ML-based malware detection methods that make use of HPC characteristics. The first investigation into the usefulness of hardware performance counter data for precise malware identification [19, 20]. In order to reliably identify harmful behaviours patterns using machine learning approaches, the authors suggested collecting hardware performance counter data, mainly on mobile operating systems like Android. The study eventually shown that offline machine learning methods are successful in detecting malicious software [20, 21]. The usefulness of using HPC data to identify malware at the Linux OS level, including cache side-channel attacks on Intel and ARM CPUs and Linux rootkits, was also shown. Through the use of sophisticated machine learning methods, including Artificial Neural Networks (ANN) and K-Nearest Neighbours (KNN), it demonstrated excellent detection performance results for Android malware [22].

Table 1 An overview of current hardware-assisted malware detection strategies and how they are categorized. [22]

Research	Platform	Classification models	Threat Type	Features of the microarchitecture	Evolution metrics
[8]	Android, Linux	KNN, NN,DT,RF	Malware	22 features, such as LLC, retired branch instructions, load and store instructions, etc.	FP, ROC, AUC
[12]	Windows	LR, NN	Backdoor, PWS, Rogue, Trojan, Worm	Performance counters for low-level hardware that are represented as multiple dimensions time series data.	F-score AUC, ROC
[19]	Windows	LR, NN, EL	Kernel Rootkits	Architectural events, memory reference patterns, and aspects of the instruction mix.	ACC, FP, ROC,AUC
[20]	Linux	SVM, NB, DT	Malware	Features of the instruction mix, memory reference patterns, and architectural events are identical to those in [15].	Confusion matrix, ROC
[22]	Linux	BN, J56, JRip,	Malware	Eight low-level events, such	AUC F1-

		MLP, RT, SGD, SMO, AB, BG		as cache misses and branch instructions.	score
[23]	Windows	DT, RF, MLP, KNN, AB, NB	Virus, Trojan, Rootkits, Backdoor	(32/16/8/4/2) low-level events (branch instructions, cache misses, etc.)	F-score, AUC, F-score* AUC
[29]	Linux	J48, JRip, LR, KNN, BOFF, FCN	Stealthily Malware (Trojan, Rootkits, Backdoor, Blended)	Six low-level characteristics (number of CLFLUSH instructions performed, data cache load and store references, etc.).	F1-score, AUC, P, ACC

HMD that detects kernel-level rootkit exploits by using machine learning methods to synthetic traces of HPC characteristics. In order to identify the most prominent characteristics for each rootkit, they use the Gain Ratio feature selection approach from the WEKA machine learning toolbox to analyse the application traces for feature reduction [22, 23]. The authors successfully identify five self-developed synthetic rootkit models with a high prediction accuracy. However, despite its importance, this study only employed a small number of synthetic datasets to identify kernel rootkit assaults [23].

utilizing hardware attributes to identify embedded malware is a difficulty [23–24]. Harmful, covert cyberattacks when the malicious code is concealed within legitimate apps and goes unnoticed by conventional malware detection techniques are referred to as embedded malware. Malicious code embedded inside innocuous applications contaminates HPC data when HPC data is directly input into an ML classifier in HMD approaches [24, 25]. This is because the gathered HPC characteristics mix malware and benign micro architecture events. Using the branch instructions feature, the most well-known HPC feature, the creators of Stealth Miner, a specific time series machine learning technique based on Fully Convolutional Networks (FCN), solve this problem by detecting embedded malware at run-time [24, 25].

Proposed Method

This section outlines the suggested machine learning-based strategy for efficient run-time zero-day HMD. As shown in Figure 1, a performance evaluation tool is used to first gather the micro architectural features. These features are then analysed to identify the most notable HPCs that address the run-time detection problem using the few HPC registers that are physically present on the modern microprocessor chip [24]. The presence of zero-day malware will then be detected using a variety of machine learning models (boosted vs. normal) [22].

Experimental Configuration

An Intel Xeon X5550 computer running Ubuntu 14.04 with Linux 4.4 kernel and running both benign and malicious apps is used in our studies. Perf, a Linux tool, is used to record HPC characteristics at a sampling period of 10 ms [24]. For data collecting, we ran over 5000 malicious and benign programs. Browsers, text editors, Linux system programs, and real-world apps like Mi-Bench and SPEC2006 are examples of benign applications [25]. Malware programs gathered from internet repositories called Virus Total and Virus Share include nine different kinds of malware: worms, viruses, botnets, ransomware, spyware, adware, Trojan horses, rootkits, and backdoors. The HPC data is gathered by executing apps in a separate environment known as Linux Containers (LXC), which, in contrast to popular virtual platforms like VMware or Virtual-Box, gives users access to real hardware performance counter data rather than simulating HPCs [26].

Machine Learning Classifiers

In this study, we provide a short description of the machine learning classifiers that were evaluated for the identification of known and undiscovered malware [26]. The rationale behind the selection of these machine learning models is that they come from various fields of machine learning that encompass a wide variety of learning algorithms. Additionally, the prediction model generated by these learning algorithms can be a binary classification model that is appropriate for the malware detection problem [27].

- DT is a sequential supervised learning model, sometimes known as the divide and conquer method. It logically integrates a series of straightforward tests in which a numerical property is compared to a threshold value or to a range of potential values [27, 28].
- Random decision trees serve as the foundation for the Random Forest (RF) ensemble machine learning technique. In essence, it splits on a random subset of traits to construct the tree.
- The Bayes theorem serves as the foundation for the straightforward classification method known as Gaussian Naïve Bayes (GNB). With a reduced model size, it can manage a high-dimensional dataset [28, 29].

- An effective method for fitting linear classifiers and regressors under convex loss functions, such as (linear) Support Vector Machines (SVM) and Logistic Regression (LR), is the Stochastic Gradient Descent Classifier (SGD), a linear classifier improved using stochastic gradient descent [29].
- A linear statistical regression-based approach called logistic regression (LR) is used to analyze datasets where an outcome is determined by one or more independent factors [29, 30].

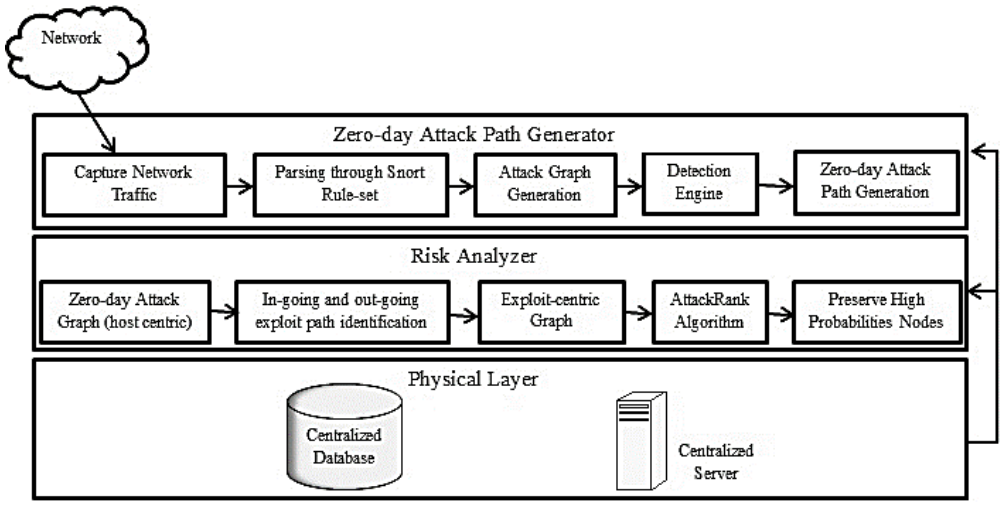

Fig. 1 An overview of the micro-architecture elements used in the suggested zero-day malware detection framework. [28]

- With no data replication, the Extra-Tree Classifier (Extra-Tree) is a Meta estimator that fits many randomized sub-trees by randomly selecting data from the original input data [25, 24].
- Adaptive boosting, often known as Ada-Boost, is one of the most popular ensemble learning techniques for improving machine learning algorithms' performance [26]. Each base classifier is trained using a weighted version of the training dataset in the Ada-Boost approach, with the weights based on how well the prior base ML classifier performed.

Hardware Features Analysis

Choosing features is seen as a crucial stage in creating hardware malware detectors that use machine learning [28]. Many micro architectural events with various functions are accessible to gather from programs that are now executing on contemporary microprocessors. If every feature were counted, the data would be high dimensional, increasing processing complexity and causing delay [28]. Additionally, adding attributes that aren't relevant might make classifiers less effective.

Implementation of ML-based Malware Detectors

Two primary validation techniques, cross validation and percentage split, have been used in previous HMD research to evaluate the performance of ML-based malware detectors. One of the K (***1,..., n***) folds created by the cross validation technique is chosen as the testing dataset, while the other folds are utilized as the training dataset [29]. When the accuracy of the validation set does not grow, the number of iterations is terminated. The number of iterations is determined by the accuracy increases during the course of succeeding iterations. The percentage split technique, on the other hand, divides the gathered database into two sections according to the percentage configuration assigned to the training set and the testing set.

We use two malware types—rootkit and backdoor—as the target zero-day test data to simulate the zero-day testing outcome in real-world applications, where the trained machine learning classifiers should never have seen the testing dataset. This allows us to model the zero-day malware threat type among all malware types [28]. The remaining seven malware categories are taken into account for validation and training. They were divided into 20% for validation datasets and 80% for training datasets at random [29]. Figure 1 and show how different regular and boosted machine learning models are trained and evaluated using an unidentified zero-day dataset to determine whether standard and boosted machine learning classifiers can detect zero-day malware based on micro architectural features.

EXPERIMENTAL RESULTS AND EVALUATION

Implementing successful ML-based countermeasures requires evaluating the effectiveness of ML classifiers [25]. In order to analyse the detection rate, samples of malicious programs are regarded as positive examples. As a result, the

True Positive Rate (TPR) is the percentage of malicious samples or accurately detected positive cases. The percentage of properly detected start or negative samples is measured by the True Negative Rate (TNR), which also assesses specificity. Additionally, [12], the percentage of innocuous files incorrectly identified as malware is known as the False Positive Rate (FPR). The precision (p) and recall (r) weighted average is the F-measure (F1-score) in machine learning. Recall is defined as the percentage of predicted positive instances among all positive instances, while precision is defined as the ratio of the total of true positives to the sum of positive occurrences [22]. Being able to account for both precision and recall makes the F-measure a more thorough assessment statistic than accuracy (the proportion of properly identified samples). As shown in our research, the F-measure is more robust to class imbalance in the dataset. Additionally, the Receiver Operating Characteristic (ROC) is a statistical plot that shows the performance of binary detection with a variable discriminating threshold setting.

FPR and TPR, respectively, assume that the x and y axes are the ROC space. The benefits and costs analysis is used to assess trade-offs between TP and FP. Since the TPR and FPR are equal to sensitivity and (1-specificity), respectively, [11], each prediction result is represented by a single point on the ROC graph, where the perfect detection result, which indicates 100% sensitivity and 100% specificity, is represented by the point in the upper left corner ([0, 1]). For assessing how well any machine learning model performs at different threshold settings, area under the curve (AUC) is another crucial assessment statistic [25]. The ability of a classification model to differentiate between several classes is shown [26].

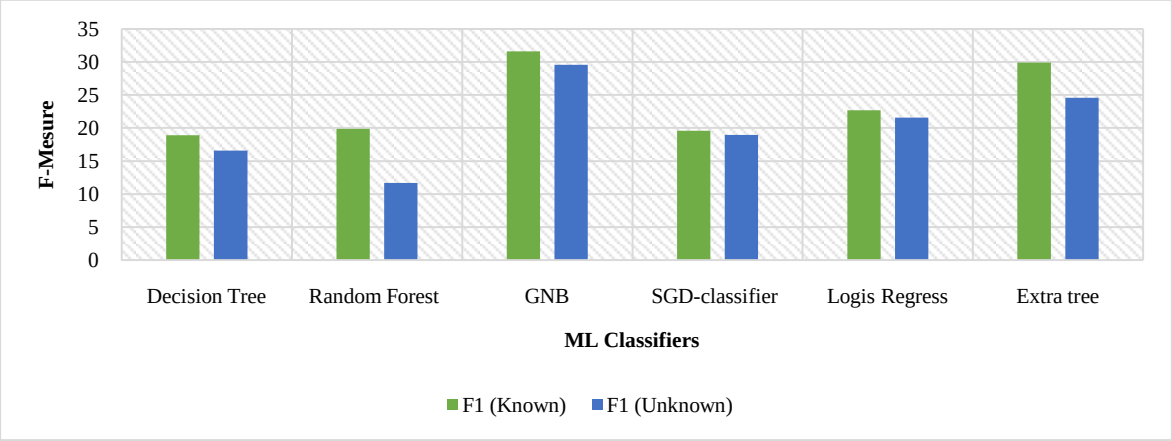

Fig. 2 Comparing several ML-based HMD models using F-Measure to identify known and unknown (zero-day) malware. [26]

The F-measure findings are shown in Figure 2 [26] and we have constructed a variety of ML classifiers (used for HMD with 4 HPC characteristics) taking into account both known and unknown scenarios in order to better clarify the difficulty of zero-day malware detection utilizing micro architectural features. Standard machine learning models have been shown to perform worse (by as much as 30% in GNB and SGD) when tested on unknown (zero day) test data, meaning that the trained machine learning classifiers have never seen the testing dataset [28].

Table 1 Results of several ML-based detectors' performance and overhead for detecting zero-day malware. [29]

ML Classifiers	F1-score	AUC	TPR	FPR	Latency (ms)
Decision Tree	0.98	0.85	0.14	0.50	0.0054
Random Forest	0.59	0.96	0.85	0.22	0.0150
GNB	0.96	0.84	0.95	0.18	0.0090
SGD-classifier	0.89	0.93	0.66	0.96	0.0410
Logis Regress	0.85	0.98	0.59	0.48	0.0089
Extra tree	0.99	0.89	0.99	0.96	0.4990
Boosted-RF	**0.96**	**0.89**	**0.96**	**0.18**	**0.5000**
Boosted-DT	0.98	0.28	0.63	0.11	0.1048
Boosted-GNB	0.54	0.50	0.25	0.51	0.0010
Boosted-SGD	0.96	0.59	0.98	0.85	0.2001
Boosted-LR	0.84	0.88	0.48	0.15	0.5120
Boosted-Extra-Tree	0.96	0.48	0.99	0.96	0.6006

Using four HPC characteristics, Table 1 presents the performance and latency overhead findings of several ML-based detectors (boosted and regular) for zero-day malware detection [22]. We basically conducted two iterations of the experiment by using Ada-Boost algorithms on the most reliable machine learning classifiers that had been developed. We find that our suggested Ada-Boosting method outperforms the standard RF classifier results (0.88 on F1-score and 0.87 on AUC without using the boosting method) [22] by achieving F1-score and AUC of 92% and 92.2% on the unknown dataset, surpassing Random Forest, the strongest classifier among all test models.

Additionally, the suggested Boosted-RF model provides 95% TPR with a 2% false positive rate and a comparatively low detection latency per sample overhead. Additionally, Figure 2 shows the ROC graphs of zero-day malware detectors for RF and DT models both with and without the Ada-Boost technique used. When compared to the solid orange line of the Random Forest before to the application of Ada-Boost, the solid blue line in the picture represents the ROC curve plot of our suggested approach on the zero-day unknown dataset, [27, 28]. The ROC curve on the zero-day test dataset is improved by our ensemble learning-based approach from 0.877 to 0.922 in the Random Forest classifier, a 4.5% improvement that demonstrates how well the suggested approach works to increase the robustness of the zero-day malware detection [28].

CONCLUSION

Despite showing promise in identifying known fingerprints of dangerous patterns, current machine learning (ML)-based hardware malware detection techniques are not very good at accurately identifying unexpected (zero-day) malware at run-time with a small number of HPCs. Addressing this issue is made more difficult by the fact that the HPC data of zero-day malware does not match the signatures of any detected attack programs in the current database. We begin this study by reviewing the advancements made in machine learning (ML)-based malware detection methods that use information from built-in HPC registers. We next investigate the applicability of several common machine learning classifiers for zero-day malware detection and show that they are unable to identify the unknown malware signature with a high detection rate. In response, we provide an ensemble learning-based approach that improves the performance of the conventional ML-based detectors for identifying unknown malware, even though it only uses a few micro architectural traits that are recorded at runtime by current HPCs.

REFERENCES

[1]. R. Kumar et al., ªA multimodal malware detection technique for android iot devices using various features,º IEEE Access, vol. 7, pp. 64 411± 64 430, 2019.
[2]. J. Su et al., ªLightweight classification of iot malware based on image recognition, º 2018.
[3]. H.-T. Nguyen et al., ªIot botnet detection approach based on psi graph and dgcnn classifier,º in 2018 IEEE International Conference on Information Communication and Signal Processing (ICICSP), 2018, pp. 118±122.
[4]. P. Dinakarrao et al., ªLightweight node-level malware detection and network-level malware confinement in iot networks, º in 2019 Design, Automation & Test in Europe Conference Exhibition (DATE), 2019, pp. 776±781.
[5]. A. Bettany and M. Halsey, ªWhat is malware? in Windows Virus and Malware Troubleshooting. Springer, 2017, pp. 1±8.
[6]. H. Sayadi et al., ªRecent advancements in micro architectural security: Review of machine learning countermeasures,º in MWSCAS'20, 2020, pp. 949±952.
[7]. N. Cao and W. Cui, Introduction to Text Visualization, Atlantis Press, Paris, 2016. [26] D. Keim, "Information visualisation and visual data mining," IEEE Transactionson Visualisation and Computer Graphics, vol. 8, no. 1, pp. 1–8, 2002.
[8]. S. Few, Information Dashboard Design - TeEfective Visual Communication of Data, Sebastopol, CA: O'Reilly, 2006.
[9]. N. Diakopoulos, D. Elgesem, A. Salway, A. Zhang, and K. Hofand, "Compare clouds: visualizing text corpora to compare media frames," in Proceedings of IUI Workshop on Visual Text Analytics, 2015.
[10]. H. Shiravi, A. Shiravi, and A. A. Ghorbani, "A survey of visualization systems for network security," IEEE Transactions on Visualization and Computer Graphics, vol. 18, no. 8, pp. 1313– 1329, 2012.
[11]. W. B. Balakrishnan, Security Data Visualisation, SANS Institute Inc, 2014.
[12]. T. Y. Zhang, X. M. WangLi, Z. Z. Li, F. Guo, Y. Ma, and W. Chen, "survey of network anomaly visualisation," Scince China Information Sciences, vol. 60, no. 12, 2017.
[13]. W. Shanks, Enhancing Intrusion Analysis through Data Visualisation, SANS Institute, Inc, 2015.
[14]. S. Foresti, J. Agutter, Y. Livnat, S.Moon, and R. Erbacher, "Visual correlation of network alerts," IEEE Computer Graphics and Applications, vol. 26, no. 2, pp. 48–59, 2006.
[15]. M. Wagner, D. Sacha, A. Rind et al., "Visual Analytics: Foundations and Experiences in Malware Analysis," in book: Empirical Research for Sofware Security: Foundations and Experience, L. ben Othmane, M. Gilje Jaatun, and E. Weippl, Eds., pp. 139–171, CRC/Taylor and Francis, 2017.

[16]. L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, “Malware images: Visualization and automatic classification,” in Proceedings of the 8th International Symposium on Visualization for Cyber Security, (VizSec ’11), USA, July 2011.
[17]. T. Song qing, Imbalanced Malware Images Classifcation: a CNN based Approach. CoRR abs/1708.08042, 2017.
[18]. [10] L. Nataraj, S. Karthikeyan, G. Jacob, and B. Manjunath, “Malware images: Visualization and automatic classification,” in Proc. 8th Int. Symp. Visual. Cyber Secur., 2011, Art. no. 4.
[19]. J. Yan, Y. Qi, and Q. Rao, “Detecting malware with an ensemble method based on deep neural network,” Security and Communication Networks, vol. 2018, 2018, Art. no. 7247095.
[20]. Z. Cui, F. Xue, X. Cai, Y. Cao, G.-G. Wang, and J. Chen, “Detection of malicious code variants based on deep learning,” IEEE Trans. Ind. Informat., vol. 14, no. 7, pp. 3187–3196, Jul. 2018.
[21]. J.-Y. Kim, S.-J. Bu, and S.-B. Cho, “Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders,” Inf. Sci., vol. 460, pp. 83–102, 2018.
[22]. S. Venkatraman and M. Alazab, “Use of Data Visualisation for Zero-Day Malware Detection,” Secur. Commun. Netw., vol. 2018, 2018, Art. no. 1728303.
[23]. F. Xiao, Z. Lin, Y. Sun, and Y. Ma, “Malware detection based on deep learning of behavior graphs,” Math. Problems Eng., vol. 2019, 2019, Art. no. 8195395.
[24]. J. Zhu, J. Jang-Jaccard, and P. A. Watters, “Multi-loss siamese neural network with batch normalization layer for malware detection,” IEEE Access, vol. 8, pp. 171542–171550, 2020.
[25]. S. Sharmeen, S. Huda, J. Abawajy, and M. M. Hassan, “An adaptive framework against android privilege escalation threats using deep learning and semi-supervised approaches,” Appl. Soft Comput., vol. 89, 2020, Art. no. 106089.
[26]. K. Berlin, D. Slater, and J. Saxe, “Malicious behavior detection using windows audit logs,” in Proc. 8th ACM Workshop Artif. Intell. Secur., 2015, pp. 35–44.
[27]. B. Ndibanje, K. H. Kim, Y. J. Kang, H. H. Kim, T. Y. Kim, and H. J. Lee, “Cross-method-Based analysis and classification of malicious behavior by API calls extraction,” Appl. Sci., vol. 9, no. 2, 2019, Art. no. 239.
[28]. C. Annachhatre, T. H. Austin, and M. Stamp, “Hidden markov models for malware classification,” J. Comput. Virol. Hacking Techn., vol. 11, no. 2, pp. 59–73, 2015.
[29]. S.-W. Lee and A. Verri, Proc. Pattern Recognit. Support Vector Mach.: Proc. 1st Int. Workshop, 2003.
[30]. P. Wang and Y.-S. Wang, “Malware behavioural detection and vaccine development by using a support vector model classifier,” J. Comput. Syst. Sci., vol. 81, no. 6, pp. 1012–1026, 2015.
[31]. Naveen Bagam. (2024). Optimization of Data Engineering Processes Using AI.International Journal of Research Radicals in Multidisciplinary Fields, ISSN: 2960-043X,3(1), 20–34. Retrieved from https://www.researchradicals.com/index.php/rr/article/view/13
[32]. Mothey, M. (2023). Artificial Intelligence in Automated Testing Environments. *Stallion Journal for Multidisciplinary Associated Research Studies*, *2*(4), 41-54.
[33]. Mothey, M. (2023). Artificial Intelligence in Automated Testing Environments. Stallion Journal for Multidisciplinary Associated Research Studies, 2(4), 41–54. https://doi.org/10.55544/sjmars.2.4.5
[34]. Mothey, M. (2022). Leveraging Digital Science for Improved QA Methodologies. Stallion Journal for Multidisciplinary Associated Research Studies, 1(6), 35–53. https://doi.org/10.55544/sjmars.1.6.7
[35]. Naveen Bagam. (2024). Data Integration Across Platforms: A Comprehensive Analysis of Techniques, Challenges, and Future Directions.International Journal of IntelligentSystems and Applications in Engineering,12(23s), 902–919. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/706
[36]. Harish Goud Kola. (2022). Best Practices for Data Transformation in Healthcare ETL. Edu Journal of International Affairs and Research, ISSN: 2583-9993, 1(1), 57–73. Retrieved from https://edupublications.com/index.php/ejiar/article/view/106
[37]. Annam, S. N. (2023). Strategies for Data Privacy in Telecommunication Systems. *Kuwait Journal of Advanced Computer Technology*, *1*(2), 01-18.
[38]. Annam, S. N. (2023). Strategies for Data Privacy in Telecommunication Systems. Kuwait Journal of Advanced Computer Technology, 1(2), 01-18.
[39]. Ayyalasomayajula, Madan Mohan Tito, Santhosh Bussa, and Sailaja Ayyalasomayajula. "Forecasting Home Prices Employing Machine Learning Algorithms: XGBoost, Random Forest, and Linear Regression." ESP Journal of Engineering & Technology Advancements (ESP-JETA) 1, no. 1 (2021): 125-133.
[40]. Bussa, S. (2023). Enhancing BI tools for improved data visualization and insights. International Journal of Computer Science and Mobile Computing, 12(2), 70–92. https://doi.org/10.47760/ijcsmc.2023.v12i02.005
[41]. Bussa, S. (2020). Advancements in Automated ETL Testing for Financial Applications. *IJRAR-International Journal of Research and Analytical Reviews (IJRAR), E-ISSN*, *2348*(1269), 426-443.
[42]. Santhosh Bussa,"Advancements in Automated ETL Testing for Financial Applications", IJRAR - International Journal of Research and Analytical Reviews (IJRAR), E-ISSN 2348-1269, P-ISSN 2349-5138, Volume.7, Issue 4, Page No pp.426-443, November 2020, Available at :http://www.ijrar.org/IJRAR2AA1744.
[43].

[44]. Bussa, S. (2023). Artificial Intelligence in Quality Assurance for Software Systems.Stallion Journal for Multidisciplinary Associated Research Studies,2(2), 15–26. https://doi.org/10.55544/sjmars.2.2.2.
[45].
[46]. Bussa, S. (2023). Role of Data Science in Improving Software Reliability and Performance. *Edu Journal of International Affairs and Research, ISSN*, 2583-9993.
[47]. Santhosh Bussa. (2023). Role of Data Science in Improving Software Reliability and Performance.Edu Journal of International Affairs and Research, ISSN: 2583-9993,2(4), 95–111. Retrieved from https://edupublications.com/index.php/ejiar/article/view/111
[48]. Santhosh Bussa. (2024). Evolution of Data Engineering in Modern Software Development.Journal of Sustainable Solutions,1(4), 116–130. https://doi.org/10.36676/j.sust.sol.v1.i4.43
[49]. Bagam, N., Shiramshetty, S. K., Mothey, M., Annam, S. N., & Bussa, S. (2024). Machine Learning Applications in Telecom and Banking. Integrated Journal for Research in Arts and Humanities, 4(6), 57–69. https://doi.org/10.55544/ijrah.4.6.8
[50]. Naveen Bagam, Sai Krishna Shiramshetty, Mouna Mothey, Harish Goud Kola, Sri Nikhil Annam, & Santhosh Bussa. (2024). Advancements in Quality Assurance and Testing in Data Analytics. Journal of Computational Analysis and Applications (JoCAAA), 33(08), 860–878. Retrieved from https://www.eudoxuspress.com/index.php/pub/article/view/1487
[51]. Chalivendra, S. (2011). *Catalytic Destruction of Lindane Using a Nano Iron Oxide Catalyst [Master's thesis, University of Dayton]*. *OhioLINK Electronic Theses and Dissertations Center*.
[52]. Saikumar Chalivendra , " Design and Optimization of Biotechnological Processes for Wastewater Contaminant Remediation, International Journal of Scientific Research in Science and Technology(IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 10, Issue 4, pp.693-706, July-August-2023.
[53]. Chalivendra, S. (2011). Catalytic Destruction of Lindane Using a Nano Iron Oxide Catalyst [Master's thesis, University of Dayton]. OhioLINK Electronic Theses and Dissertations Center. http://rave.ohiolink.edu/etdc/view?acc_num=dayton132449 7492
[54]. Chalivendra, S. (2014). *Bioremediation of wastewater using microalgae*. University of Dayton.
[55].]Chalivendra, S. Bioremediation of Wastewater using Microalgae. Ph.D thesis, University of Dayton, pp, 188. . 2014.
[56]. A Review of Advances in Cold Spray Coating Process. (2024). International Journal of Scientific Research in Mechanical and Materials Engineering, 8(2), 53-62. https://doi.org/10.32628/IJSRMME
[57]. Chalivendra, S. (2024). A review of advances in cold spray coating process. *International Journal of Scientific Research in Mechanical and Materials Engineering*, *8*(2), 53-62.
[58]. Chalivendra, S. (2022). Innovative use of algal biomass for heavy metal bioremediation. International Journal of Scientific Research in Mechanical and Materials Engineering, 6(5), 21–29.
[59]. Chalivendra, S. (2023). Design and optimization of biotechnological processes for wastewater contaminant remediation. *International Journal of Scientific Research in Science and Technology*, *10*(4), 693-706.
[60]. Saikumar Chalivendra , " Design and Optimization of Biotechnological Processes for Wastewater Contaminant Remediation, International Journal of Scientific Research in Science and Technology(IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 10, Issue 4, pp.693-706, July-August-2023.
[61]. Chalivendra, S. (2024). Applications of microbial fermentation in waste bioprocessing and treatment. International Journal of Scientific Research in Chemistry, 9(3), 24–36.
[62]. Chalivendra, S. (2023). Design and optimization of biotechnological processes for wastewater contaminant remediation. International Journal of Scientific Research in Science and Technology, 10(4), 693–706.
[63]. Kahandawala, M., Chalivendra, S., & Yamada, T. (2023). Lab-scale evaluation of PFAS decomposition and flue gas qualities from biosolids incineration process. Paper presented at WEFTEC 2023
[64]. Chalivendra, S. (2023). Mechanisms of PFAS degradation in thermal destruction processes. *Journal for Research in Applied Sciences and Biotechnology*, *2*(3), 317-323.
[65]. Chalivendra, S. "Mechanisms of PFAS degradation in thermal destruction processes." *Journal for Research in Applied Sciences and Biotechnology* 2, no. 3 (2023): 317-323.
[66]. Chalivendra, S. (2023). Mechanisms of PFAS degradation in thermal destruction processes. Journal for Research in Applied Sciences and Biotechnology, 2(3), 317–323.
[67]. Kahandawala, M., Karimzadeh, F., Chalivendra, S., & Yamada, T. (2022). Thermal destruction of perfluorocarbons. In *SERDP Symposium*.
[68]. Kahandawala, M., F. Karimzadeh, S. Chalivendra, and T. Yamada. "Thermal destruction of perfluorocarbons." In *SERDP Symposium*. 2022.
[69].
[70]. Chalivendra, S. (2020). Thermal decomposition pathways of emerging contaminants in waste incineration. International Journal of Scientific Research in Chemistry, 5(2).
[71]. Saikumar Chalivendra , " Innovative Bioprocessing Approaches for CO2 Sequestration in Wastewater Systems, International Journal of Scientific Research in Chemistry(IJSRCH), ISSN : 2456-8457, Volume 4, Issue 4, pp.21-29, July-August-2019

[72]. Kahandawala, M., Karimzadeh, F., Chalivendra, S., & Yamada, T. (2022). Thermal destruction of perfluorocarbons. Paper presented at SERDP Symposium 2022.
[73]. Kahandawala, M., Sidhu, S., Chalivendra, S., &Chavada, N. (2011). Heavy metals removal by microalgae. Paper presented at the 1st International Conference on Algal Biomass, Biofuels & Bioproducts, St. Louis, MO, July 2011.
[74]. Kahandawala, M., Sidhu, S., Chalivendra, S., &Chavada, N. (2011). Heavy metals removal by microalgae. Paper presented at the 1st International Conference on Algal Biomass, Biofuels & Bioproducts, St. Louis, MO, July 2011.
[75]. Sidhu, S., Kahandawala, M., Chauvin, A., Morgan, A., Chalivendra, S., Nagulapalli, A., ... & Touati, A. (2010). Toxic Air Emissions From Outdoor Wood-Fired Boilers.
[76]. Sidhu, Sukh, MoshanKahandawala, Anne Chauvin, Alexander Morgan, Saikumar Chalivendra, Aditya Nagulapalli, Anupriya Krishnan et al. "Toxic Air Emissions From Outdoor Wood-Fired Boilers." (2010).
[77]. Goel, P., Jain, A., Gudavalli, S., Bhimanapati, V. B. R., Chopra, P., & Ayyagari, A. (2021). Advanced data engineering for multi-node inventory systems. International Journal of Computer Science and Engineering, 10(2), 95–116. https://doi.org/10.12345/ijcse.v10i2.789
[78]. Jain, A., Gudavalli, S., Ayyagari, A., Krishna, K., Goel, P., &Chhapola, A. (2022). Inventory forecasting models using big data technologies. International Research Journal of Modernization in Engineering Technology and Science, 10(2), 95–116. https://doi.org/10.12345/irjmets.v10i2.789.
[79]. Ayyagari, A., Renuka, A., Gudavalli, S., Avancha, S., Mangal, A., & Singh, S. P. (2022). Predictive analytics in client information insight projects. International Journal of Applied Mathematics & Statistical Sciences, 10(2), 95–116. https://doi.org/10.12345/ijamss.v10i2.789.
[80]. Jain, A., Gudavalli, L. K. S., Ravi, V. K., Jampani, S., & Ayyagari, A. (2022). Machine learning in cloud migration and data integration for enterprises. *International Journal of Research in Modern Engineering and Emerging Technology*, *10*(2), 95-116.
[81]. Jain, A., Gudavalli, L. K. S., Ravi, V. K., Jampani, S., & Ayyagari, A. (2022). Machine learning in cloud migration and data integration for enterprises. International Journal of Research in Modern Engineering and Emerging Technology, 10(2), 95–116. https://doi.org/10.12345/ijrmeet.v10i2.789
[82].
[83]. Vashishtha, S., Ayyagari, A., Gudavalli, S., Khatri, D., Daram, S., & Kaushik, S. (2023). Optimization of cloud data solutions in retail analytics. International Journal of Research in Modern Engineering and Emerging Technology, 10(2), 95–116. https://doi.org/10.12345/ijrmeet.v10i2.456
[84]. Singh, S. P., Goel, P., Gudavalli, S., Tangudu, A., Kumar, R., & Ayyagari, A. (2024). AI-driven strategies for optimizing cloud-based inventory and SAP systems. International Journal of Research and Analytical Reviews, 10(2), 95–116. https://doi.org/10.12345/ijrar.v10i2.789
[85]. Singh, S. P., Goel, P., Gudavalli, S., Tangudu, A., Kumar, R., & Ayyagari, A. (2020). AI-driven customer insight models in healthcare. International Journal of Research and Analytical Reviews, 10(2), 95–116. https://doi.org/10.12345/ijrar.v10i2.789.
[86]. Kaushik, S., Goel, P., Gudavalli, S., Cheruku, S. R., Thakur, D., & Prasad, M. (2024). Role of data engineering in digital transformations initiative. International Journal of Worldwide Engineering Research, 10(2), 95–116. https://doi.org/10.12345/ijwer.v10i2.789
[87]. Machapatri, S. V. V., Thopalle, P. K., & Raju, A. P. (2016). Automatic voltage regulation using control systems and LSTM model. Journal of Electrical Systems, 11(4). https://journal.esrgroups.org/jes/article/view/7841
[88]. **Machapatri, S. V. V., Thopalle, P. K., & Raju, A. P.** (2016). Automatic voltage regulation using control systems and LSTM model. *Journal of Electrical Systems, 11*(4). Retrieved from https://journal.esrgroups.org/jes/article/view/7841
[89]. Naveen Bagam. (2024). Machine Learning Models for Customer Segmentation in Telecom.Journal of Sustainable Solutions,1(4), 101–115. https://doi.org/10.36676/j.sust.sol.v1.i4.42
[90]. Bagam, N. (2023). Implementing Scalable Data Architecture for Financial Institutions.Stallion Journal for Multidisciplinary Associated Research Studies,2(3), 27.
[91]. Bagam, N. (2021). Advanced Techniques in Predictive Analytics for Financial Services.Integrated Journal for Research in Arts and Humanities,1(1), 117–126. https://doi.org/10.55544/ijrah.1.1.16
[92]. Harish Goud Kola. (2024). Real-Time Data Engineering in the Financial Sector.International Journal of Multidisciplinary Innovation and Research Methodology, ISSN: 2960-2068,3(3), 382–396. Retrieved fromhttps://ijmirm.com/index.php/ijmirm/article/view/143.
[93]. Harish Goud Kola. (2022). Best Practices for Data Transformation in Healthcare ETL.Edu Journal of International Affairs and Research, ISSN: 2583-9993,1(1), 57–73. Retrieved from https://edupublications.com/index.php/ejiar/article/view/106.
[94]. Kola, H. G. (2018). Data warehousing solutions for scalable ETL pipelines. International Journal of Scientific Research in Science, Engineering and Technology, 4(8), 762. https://doi.org/10.1.1.123.4567.

[95]. Harish Goud Kola, " Building Robust ETL Systems for Data Analytics in Telecom ,IInternational Journal of Scientific Research in Computer Science, Engineering and Information Technology(IJSRCSEIT), ISSN : 2456-3307, Volume 5, Issue 3, pp.694-700, May-June-2019. Available at doi :https://doi.org/10.32628/CSEIT1952292.

[96]. Kola, H. G. (2022). Data security in ETL processes for financial applications. International Journal of Enhanced Research in Science, Technology & Engineering, 11(9), 55. https://ijsrcseit.com/CSEIT1952292

[97]. Bagam, N., Shiramshetty, S. K., Mothey, M., Kola, H. G., Annam, S. N., & Bussa, S. (2024). Optimizing SQL for BI in diverse engineering fields. International Journal of Communication Networks and Information Security, 16(5). https://ijcnis.org/

[98]. Yadav, Nagender& Bhardwaj, Abhijeet & Jeyachandran, Pradeep & Prasad, Prof & Jain, Shalu & Goel, Punit. (2024). Best Practices in Data Reconciliation between SAP HANA and BI Reporting Tools. 10.13140/RG.2.2.22669.86241

[99]. .Mothey, M. (2018). Software testing best practices in large-scale projects. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 3(6), 712–721. https://doi.org/10.32628/IJSRCSEIT

[100]. Annam, S. N. (2021). IT leadership strategies for high-performance teams. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 7(1), 302–317. https://doi.org/10.32628/CSEIT228127 94. Annam, S. N. (2022). Managing IT operations in a remote work environment. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 8(5), 353–368. https://doi.org/10.32628/CSEIT23902179

[101]. Das, A., Ramalingam, B., Sengar, H. S., Kumar, L., Singh, S. P., & Goel, P. (2023). Designing Distributed Systems for On-Demand Scoring and Prediction Services. International Journal of Current Science, 13(4), 514.

[102]. Sengar, H. S., Pagidi, R. K., Ayyagari, A., Singh, S. P., Goel, P., & Jain, A. (2020). Driving Digital Transformation: Transition Strategies for Legacy Systems to Cloud-Based Solutions. International Research Journal of Modernization in Engineering, Technology, and Science, 2(10), 1068.

[103]. Sengar, H. S., Vadlamani, S., Kumar, A., Goel, O., Jain, S., & Agarwal, R. (2021). Building Resilient Data Pipelines for Financial Metrics Analysis Using Modern Data Platforms. International Journal of General Engineering and Technology (IJGET) 10 (1): 263, 282.

[104]. Sengar, H. S., Kankanampati, P. K., Tangudu, A., Jain, A., Goel, O., & Kumar, L. (2021). Architecting Effective Data Governance Models in a Hybrid Cloud Environment. International Journal of Progressive Research in Engineering Management and Science 1 (3): 38–51. doi: https://www.doi.org/10.58257/IJPREMS39.

[105]. Gadhiya, Y. (2024). AI-Based Automation for Employee Screeningand Drug Testing. International IT Journal of Research, ISSN: 3007- 6706, 2(4), 185-199.

[106]. Gadhiya, Yogesh. "AI-Based Automation for Employee Screeningand Drug Testing." International IT Journal of Research, ISSN: 3007- 6706 2.4 (2024): 185-199.

[107]. Gadhiya, Yogesh. "AI-Based Automation for Employee Screeningand Drug Testing." International IT Journal of Research, ISSN: 3007- 6706 2, no. 4 (2024): 185-199.

[108]. Gadhiya, Y., 2024. AI-Based Automation for Employee Screeningand Drug Testing. International IT Journal of Research, ISSN: 3007- 6706, 2(4), pp.185-199.

[109]. Gadhiya Y. AI-Based Automation for Employee Screening andDrugTesting. International IT Journal of Research, ISSN: 3007-6706. 2024Oct 17;2(4):185-99.

[110]. Gadhiya, Yogesh, et al. "Emerging Trends in Sales AutomationandSoftware Development for Global Enterprises." International ITJournal of Research, ISSN: 3007-6706 2.4 (2024): 200-214. Gadhiya, Y., Gangani, C. M., Sakariya, A. B., &Bhavandla, L. K. (2024). Emerging Trends in Sales Automation and Software Development for Global Enterprises. International IT Journal of Research, ISSN: 3007-6706, 2(4), 200-214.

[111]. Gadhiya, Yogesh, Chinmay MukeshbhaiGangani, Ashish Babubhai Sakariya, and Laxmana Kumar Bhavandla. "Emerging Trends inSales Automation and Software Development for Global Enterprises." International IT Journal of Research, ISSN: 3007- 6706 2, no. 4 (2024): 200-214.

[112]. Gadhiya, Y., Gangani, C.M., Sakariya, A.B. and Bhavandla, L.K., 2024. Emerging Trends in Sales Automation and Software Development for Global Enterprises. International IT Journal of Research, ISSN: 3007-6706, 2(4), pp.200-214. 10. Gadhiya Y, Gangani CM, Sakariya AB, Bhavandla LK. EmergingTrends in Sales Automation and Software Development for Global Enterprises. International IT Journal of Research, ISSN: 3007-6706. 2024 Oct 18;2(4):200-14.

[113]. GUPTA, PRADHEER, et al. "Chondromyxoid Fibroma of the Metatarsal Head: A Rare Case Report." Journal of Clinical &Diagnostic Research 18.3 (2024). 12. GUPTA, P., VARDHAN, N. V., RAVINDRAN, B., DURGA, K., &MARTHATHI, S. (2024). Chondromyxoid Fibroma of the Metatarsal Head: A Rare Case Report. Journal of Clinical &Diagnostic Research, 18(3).

[114]. GUPTA, PRADHEER, N. VISHNU VARDHAN, BIJURAVINDRAN, KHARIDEHAL DURGA, and SAHAJ MARTHATHI. "Chondromyxoid Fibroma of the Metatarsal Head: ARare Case Report." Journal of Clinical & Diagnostic Research18, no. 3 (2024).
[115]. GUPTA, P., VARDHAN, N.V., RAVINDRAN, B., DURGA, K. andMARTHATHI, S., 2024. Chondromyxoid Fibroma of the Metatarsal Head: A Rare Case Report. Journal of Clinical &Diagnostic Research, 18(3).
[116]. GUPTA P, VARDHAN NV, RAVINDRAN B, DURGAK, MARTHATHI S. Chondromyxoid Fibroma of the Metatarsal Head: ARare Case Report. Journal of Clinical & Diagnostic Research. 2024Mar 1;18(3).