IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 22, Issue 4, Ser. II (Jul. – Aug. 2020), PP 14-20
www.iosrjournals.org

An Investigation for Mobile Malware Behavioral and Detection Techniques Based on Android Platform

Amira B. Sallow[1], Mohammed A. M.Sadeeq[2], Rizgar R. Zebari[3], Maiwan B. Abdulrazzaq[4], Mayyadah R. Mahmood[5],Hanan M. Shukur[6], Lailan M. Haji[7]

[1]Computer Science & IT, college of Science, Nawroz University, Kurdistan region – Iraq
[2]Duhok Polytechnic University Presidency, Duhok, Kurdistan region – Iraq
[3]IT Department,Duhok Polytechnic University, Duhok, Kurdistan region – Iraq
[4]Computer Science dept., college of Science/ University of Zakho, Kurdistan region – Iraq
[5]Computer Science dept., college of Science/ University of Zakho, Kurdistan region – Iraq
[6]IT Department, Al Kitab University, Kirkuk - Iraq
[7]Computer Science dept., college of Science/ University of Zakho, Kurdistan region – Iraq

Abstract

Nowadays, with portable computation equipment becoming extra commonly depended besides incorporated through extra sensitive structures such as infrastructure besides armed or regulation implementation. However, is being clearer when considering that malware and other types with cyberattacks attack such portable structures. Malware has always been an issue regarding any technological advances in the world of software. Smartphones and other mobile devices are thus to be expected to face the same problems. In this paper, mobile malware detection techniques presented in details. Furthermore, we have stated the recent techniques of mobile malware detection. Adding to that, these closest works to the paper objective are compared with respect to the based-portal, depended-approach, and significant system's description points.

***Key Word**: malware, cyberattacks, android malware, malware detection, mobile application.*

Date of Submission: 17-07-2020 Date of Acceptance: 01-08-2020

I. Introduction

Android has been developed and designed to be an open-source framework for smartphone apps. Android applications used high-level technologies in both software and hardware fields and user data and information stored in local and server devices uncovered by the operating system to provide the consumers with value and innovation [1]–[3]. As an Open source platform, Android should have strong security mechanism to ensure the security of the user application, information, and data with severe security architecture [2]–[4].

Malware attacks on smartphones have occurred in recent days due to private information leakage and lead the system to crash the entire phone system. Malicious code is inserted in most of the application to steal other's data. The malware attacks on every phone in this way[4]–[7]. The malware decreases the assurance of system information by operating system weakness [8]–[10]. Once inserted into the smartphone the malware gains root access to each file. The rooting can permanently damage the device[11]–[14]. Detection of malware is a critical task, because malware developers hide their malicious actions and implement novel techniques to prevent detection [15]–[17].

In this, paper a wide-ranging analysis of literature with other publications that have used malware detection strategies and discussion with their algorithms over the last five years. The manuscript's remaining-parts are literature survey, discussion and recommendations, and the conclusion.

II. Investigation for Web Attacks and Their Effects on Mobile Networks

Malicious SWcan be considered asevery programming deliberatelyintendedcausingharming PC, Client/Server, workstation, or Mobile device[18]–[22]. There is an extensivecategoriesdiversity for malware, comprising computer infections.PC codesas well reflected malware whenfurtivelydeedin contradiction of the PC-benefitsoperator[23]–[27]. Collectionof unwilling the operation of virusesdepended forassistancedefendin contradiction of the malware activities, whichaids fordistinguishingthat previouslyexisted, also forrecuperating from malwaremovements.Mobile malware speedilyunattractiveasrealhazard[28]–[32]. It may telltaleagainstthe phone. Transcribedvia committed causingdestruction.Typical mobile risk comprises 3 kinds (mal, gray and spy)-wares. These 3 kinds can be differentiated depending on transferringapproach, validity, and user-sign[33]–[37].

Network messaging describes regular procedures guidelines/morals permitusedagendasforconversationtogether deprived ofHW respectbesides dependedOSs. Moveable networks communicateviawireless medium, whichspreadin excess of cells.All these cells assistedviaa static position transmitter or more named standard transmitter centers[38], [39]. The standard centersaffordlink exposure to cellsassound broadcasting andinformation. Various frequencies depended foradjacent cells in order to evadeintrusionwithdeliveringassuredfacilityclassinsideevery cells.Cells-intersectionofferwirelessexposurethrougheclectictopographicalzone. Hence, capable offrequentmoveable transmitters fortogether-intercommunication viaphoneswherever insidesystem[40].

Cellular networks have variouscharacteristics: Extracapabilityin compare toparticularbig transmitter, Moveableexpedientsspendfewerenergy in compared toparticular transmitter, Greaterexposurezonewith respect toparticularearthly transmitter.Mainbroadcastingsbenefactorsconsumeorganizedspeechbesidesinformation cellular netsaboveutmostoccupiedterrestrialzone of Ground. However,moveabledevices permit moveablecalculating equipment for connectionthroughcommunityconvertedphonenets alsocommunal Internet[41].

III. Malware Detection Techniques

Nowadays, the most standard operating system for smartphone devices is Android as a result of that the malware numbers have been increased compared with the previous years[42]–[46]. Many anti-malware programs have developed to protect personal and sensitive user information stored in mobile from such attacks [11], [47]–[50]. Moreover, Android malware has been on a rising trend due to the prevalence of the Android operating system. Without user permission, and explicitly prompting users, Android malware is installed and run on mobile phones that pose significant threats to users, such as personal information leakage and advanced fraud. Researchers and practitioners are suggesting various techniques to tackle these threats[51]–[54]. Static analysis is one of these techniques which is widely applied to detect Android malware and prevent it before installation [55]–[59].

Malware is among the big problems in the operating system or the world of software. Android also endures the same difficulties. Signature-based malware detection techniques were also employed to detect malware [60]–[64]. It is still a crucial issue to accurately detection any new malware despite numerous detection and analysis techniques [41]–[43]. Malicious applications are a significant threat to the cybersecurity of the increased mobile devices and their user numbers. So, trying to detect malicious Android apps is a difficult task given the huge count of Android apps and their properties that have a wide range of features and a limited dataset [68]–[72]. Malware is a software that has malicious behavior has designed intentionally to penetrate or damage a computer device without the owner's permission. Malware detection relates to the procedure of finding malware on a host device or determining whether a particular program is malicious or benign [73]–[75]. An anti-malware automatically distinguishes malicious software from benign applications to avoid computer damage. Eventually, once malware has detected, it is disabled by anti-malware to prevent it from infecting the computer and executing it. There are mainly three cases of use which drive malware analysis [76]:

- Computer security incident management: If an organization discovers or distrusts that, some malware may have entered its systems. A response team may want to conduct malware analysis on any possible samples discovered during the search process to determine whether they are malware. If that is so, what effect this malware will have on the systems within the environment of the target organizations.

- Malware researchers: researchers from academia or industry may perform malware analysis primarily to identify how malware is operating and the latest techniques used in its development.

- Indicator of compromise extraction: Software product and service providers may do bulk malware analysis to identify possible new vulnerability compromise indicators; this information will then feed the security product or solution to helping clients had better protect themselves from malicious attacks.

IV. Literature Survey

Khatri and Abendroth[15], in 2015, demonstrated network malware detection system as mobile guard and explain its key features, as well as protecting end-users from malware attacks propagated through the network of mobile operators. they simulated the mobile guard using demonstration data, send the malicious traffic to the mobile guard instance, and show the identification results of malicious events on the dashboard.

Wang et al. [77] in 2015, Suggested to use Cuckoo Droid an open source framework as a hybrid detection system that used the features of Cuckoo Sandbox that use static and dynamic analysis to analyze Android malware. The proposed framework consists of two components: the anomaly detection engine and signature-based detection engine. The first component uses dynamic analysis to anomalous applications while the other component uses a combination of static and dynamic analysis to detect and classify known malware.

Park et al. [78]in 2015, the android malicious device identification method was launched using static analysis and malicious function similarities. Their system intercepts, collects, and analyzes the malicious code, rather than the device. It can prevent the installation of malicious smartphone application and analyze it quickly and accurately. In addition, ranges of malicious applications that attack the Android operating system are close to known malware and repackaged a malicious application that existed. It provides a special feature that can be associated with cumulative malware in the downloaded applications.

Wen and Yu. [79]in 2017, proposed a lightweight learning machine, able to identify malware on Android devices. The features were derived using the static and the dynamic analyses, A new feature selection approach is then presented based on the Principle Component Analysis (PCA) to use to reduce the overall feature dimensions, then support vector machine (SVM) was used for malware classification. Experimental tests reveal that the device is suggested has an efficient tool for detecting malware in Android.

Liang, et al. [80] in 2017, an end-to-end deep learning platform for detecting Android malware was introduced. Their model takes the raw call sequence of the device that is created during runtime of the application as input and determines if the sequence is malicious without any manual interference. The model evaluation was conducted on 14231 Android apps and obtained a 93.16 percent detection accuracy, which is 2.81 percent higher than the contrast experiment in which other researchers implemented the method proposed.

Ali et al. [65] in 2017, an anomaly-based malware identification mechanism has been suggested for an Android operating system. To analyze the behavioral patterns. They used a dataset that was installed on Android device and consists of two types of applications (benign, malicious). In addition, execute it in a controlled environment generating the system metrics (feature vector) from each app. A range of machine learning algorithms are then used to classify the application as benign or malware: Tree of Decision, K Nearest Neighbor, Logistic Regression, Multilayer Perceptron Neural Network, Naive Bayes, Random Forest, and Vector Machine Support. Each algorithm was assessed using different performance parameters to determine which ones are better suited for malicious detection.

Amro[81] in 2018, proposed a malware detection technique called Personal Mobile Malware Guard – PMMG- that classifies malwares based on the mobile user feedback. PMMG controls permissions of different applications and their behavior according to the user needs. These preferences are built incrementally on a personal basis according to the feedback of the user. Performance analysis showed that it is theoretically feasible to build PMMG tool and use it on mobile devices.

Liu et al. [82] in 2018, investigated the use of evolutionary computing techniques, both to automatically develop new variants of mobile malware that successfully evade static-based anti-malware systems, and to develop better security solutions against them. Coevolutionary process of arms race has also been considered a possible candidate for building a more reliable system against new threats and testing of the system.

Fan et al. [83]in 2018, addressed in what way to use both content-based and relation-based features for characteristically malware, and to construct different forms of entities and their rich semantic relationships, structural and present meta-graph-based information network (HIN) were built. To calculate connectivity over built-in HIN files, as malware detection is cost-sensitive, efficient methods are needed to learn latent HIN representations. To address this challenge, a new HIN embedding model metagraph2with based on the built meta-graph schemes at the first attempt to learn the low-dimensional representations for HIN nodes, where all HIN structures and semantics are preserved for the detection of malware. Experimental results show that other alternate malware detection approaches have been underperformed by the Scorpion framework that implements the proposed method.

Sen et al. [84] in 2018, proposed a high efficiency Android malware hybrid detection scheme. The scheme employed various methods of analysis (static and dynamic approaches) to construct a robust detection scheme. Com+ feature as suggested detection approach that Based on the classic permission and API call features to enhance the productivity of static detection. The failure of the classic function call-based malware detection issue has also been prevented as a result of the adoption of function selection and clustering method to unify function call graph features of different dimensions into the same dimension. The results of the experiments showed that the proposed system achieved high accuracy in malware detection, and the proposed system could be used to establish cloud services for Android malware detection.

Shen et al. [85] in 2019, developed a mobile malware detection technique based on the analysis of the information flows. The suggested method explores the design of information flows to determine behavior patterns present in them and which flows are related, that share partial processing paths. N-gram analyzes are used to classify special and typical patterns of activity found in Complex-Flows. The N-gram analysis is carried out on sequences of API calls that occur along the control flow paths of Complex-Flows. By applying it to four different data sets totaling 8,598 apps, the result showed the precision of the proposed technique.

Pang and Bian[86], in suggested a static detection method based on Naive Bayes forAndroid malware. Requested permissions, system API calls, and activity ratio were extracted through Android packages from the four major Android components. Three types of information were used as features for characterizing each application, and through Naïve Bayes classifier perform classification model training and malware detection. The suggested solution did not run Android applications, which decreases the the experiment cost and completes malware identification before installation by the customers.

Sabhadiya et al. [87] in 2019, aimed at implementing a deep learning model that can automatically detect whether or not an Android application is infected with malware without installation by examining the different android malware and its methods based on various deep learning android malware detection techniques such as Maldozer, DroidDetector, DroidDeepLearner, DeepFlow, DroidDelver, and Droid Deep. The summary of all previous literature review is illustrated in Table 1.

Table 1: Comparison of the Reviewed Researches Related to Malware Detection

Author(s)	Year	Platform	Approach	Description
Khatri and Abendroth[15]	2015	Android	Mobile Guard	Network system created to support mobile networks to distinguish malicious actions and take needed activities to avoid malicious behaviors and frauds
Wang et al. [77]	2015	Android	Hybrid Detection System	The proposed framework consists of two components: the anomaly detection engine and signature-based detection engine. The first component uses dynamic analysis to anomalous applications while the other component uses a combination of static and dynamic analysis to detect and classify known malware.
Park et al. [78]	2015	Android	Static Analysis	Presents a unique feature that can be compared to accumulated malware in downloaded applications.
Wen and Yu [79]	2017	Android	Machine Learning	The features were derived using the static and the dynamic analyses; followed by a new feature selection technique based on the Principle Component Analysis (PCA) to use to reduce the overall feature dimensions, then support vector machine (SVM) was used for malware classification.
Liang et al. [80]	2017	Android	Deep Learning	The model takes as input the raw call sequence of the device created during the application runtime, and determines whether the sequence is malicious without manual interference.
Ali et al. [65]	2017	Android	Machine Learning	Used a dataset that was installed on Android device and consists of two types of applications (benign, malicious).
Amro[81]	2018	Android	Personal Mobile Malware Guard	classifies malwares based on the mobile user feedback
Liu et al. [82]	2018	Android	Evolutionary Computation	Exploring the use of evolutionary computational methods for improving the Anti-malware and Mobile Malware.
Fan et al. [83]	2018	Android	Machine Learning	Develop Scorpion 's intelligent Detection System. in Scorpion, at the first attempt to learn the low-dimensional representations for HIN nodes, where all HIN structures and semantics are preserved for the detection of malware.
Sen et al. [84]	2018	Android	Hybrid Detection System	Com+ feature as suggested detection approach that Based on the classic permission and API call features to enhance the productivity of static detection.
Shen et al. [85]	2019	Android	Complex-Flows	Complex-Flows a new information flows representation was used to obtain application behavior on device sensitive data.
Pang and Bian[86]	2019	Android	Naive Bayes	Naive Bayes classifier has been used to detect malware by classifying three types of data (system API calls, the ap, requested permissions) that have been extracted using static analysis.
Sabhadiya et al. [87]	2019	Android	Deep Learning	Numerous deep learning methods have been used to detect different types of Android malware detection techniques.

V. Discussion

The main findings from the literature are summarized in Table 1. Note that most of the approaches were built to detect malware behavior for android applications, some of them used machine learning to classified the app as benign or malware ([79], [65], [83]), while other use deep learning to detect the malware ([80], [87]). Moreover, the combination of static and dynamic analysis used to perform detection and classification of known malware ([77], [84]) while others used static ([78]). Some researchers build Demonstrated Mobile Guard as a malware detection system to protect users from malware attacks ([15]). In addition, other researchers tried different Approaches such as evolutionary computation ([82]), Naive Bayes ([86]), Complex-Flows ([85]).

VI. Recommendations

After reviewing different Android malware detection techniques for the recent five years for thirteen different works, machine learning is recommended as a widely used technique to classified the app as benign or malware.

VII. Conclusion

Android is open source, and any changes are easy to make. Thus, attackers can add different security vulnerabilities to the Android platform. With this development of Android smartphones and the number of Android applications, as well as malware, it is expanding daily and malware detection in the Android platform is becoming a critical requirement to defend mobile users against attackers and a lot of effective malware detection techniques have been developed in the last few years. This paper discusses recent research works over the past five years on malware detection strategies for the Android mobile operating system.

References

[1] D. B. Kirk and W. H. Wen-mei, *Programming massively parallel processors: a hands-on approach. Newnes*. Oxford, 2012.

[2] M. B. Abdulrazaq and O. M. Mustafa, "Designing and Implementing of An Online Library Managment System," *Science Journal of University of Zakho*, vol. 5, no. 3, Art. no. 3, Sep. 2017, doi: 10.25271/2017.5.3.396.

[3] S. Q. Sabri, A. M. Ahmad, and M. B. Abdulrazaq, "Design and Implementation of Student and Alumni Web Portal," *Science Journal of University of Zakho*, vol. 5, no. 3, pp. 272–277, 2017.

[4] A. A. Salih and M. B. Abdulrazaq, "Combining best features selection using three classifiers in intrusion detection system," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 94–99.

[5] H. I. Dino and M. B. Abdulrazzaq, "Facial Expression Classification Based on SVM, KNN and MLP Classifiers," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 70–75.

[6] M. Abdulrazaq and A. Salih, "Combination of multi classification algorithms for intrusion detection system," *Int. J. Sci. Eng. Res.*, vol. 6, no. 1, pp. 1364–1371, 2015.

[7] M. B. Abdulrazzaq and J. N. Saeed, "A Comparison of Three Classification Algorithms for Handwritten Digit Recognition," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 58–63.

[8] S. R. Zeebaree, K. F. Jacksi, and R. R. Zebari, "Impact analysis of SYN flood DDOS attack on HAPROXY and NLB cluster-base web servers," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 1, Art. no. 1, Jul. 2020, doi: 10.11591/ijeecs.v19.i1.pp%p.

[9] R. R. Zebari, S. R. Zeebaree, and K. Jacksi, "Impact Analysis of HTTP and SYN Flood DDoS Attacks on Apache 2 and IIS 10.0 Web Servers," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 156–161.

[10] S. R. Zeebaree, R. R. Zebari, and K. Jacksi, "Performance analysis of IIS10.0 and Apache2 Cluster-based Web Servers under SYN DDoS Attack," *TEST Engineering & Management*, vol. 83, no. March-April 2020, pp. 5854–5863, 2020.

[11] G. Shanmugasundaram, S. Balaji, and T. Mugilan, "Investigation of Malware Detection Techniques on Smart Phones," in *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA)*, Pondicherry, Jul. 2018, pp. 1–4, doi: 10.1109/ICSCAN.2018.8541197.

[12] M. B. Abdulrazzaq and K. I. Khalaf, "Handwritten Numerals' Recognition in Kurdish Language Using Double Feature Selection," in *2019 2nd International Conference on Engineering Technology and its Applications (IICETA)*, 2019, pp. 167–172.

[13] M. R. Mahmood and A. M. Abdulazeez, "A Comparative Study of a New Hand Recognition Model Based on Line of Features and Other Techniques," in *International Conference of Reliable Information and Communication Technology*, 2017, pp. 420–432.

[14] M. R. Mahmood, A. M. Abdulazeez, and Z. Orman, "Dynamic Hand Gesture Recognition System for Kurdish Sign Language Using Two Lines of Features," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 42–47.

[15] V. Khatri and J. Abendroth, "Mobile Guard Demo: Network Based Malware Detection," in *2015 IEEE Trustcom/BigDataSE/ISPA*, Helsinki, Finland, Aug. 2015, pp. 1177–1179, doi: 10.1109/Trustcom.2015.501.

[16] R. Zebari, S. Zeebaree, K. Jacksi, and H. Shukur, "E-Business Requirements for Flexibility and Implementation Enterprise System: A Review," *International Journal of Scientific & Technology Research*, vol. 8, pp. 655–660, Nov. 2019.

[17] M. A. Mohammed *et al.*, "An anti-spam detection model for emails of multi-natural language," *Journal of Southwest Jiaotong University*, vol. 54, no. 3, 2019.

[18] O. M. Ahmed and A. B. Sallow, "Android security: a review," *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 135–140, 2017.

[19] C. Andri, M. HazimAlkawaz, and A. Bibo Sallow, "Adoption of Mobile Augmented Reality as a Campus Tour Application," *IJET*, vol. 7, no. 4.11, p. 64, Oct. 2018, doi: 10.14419/ijet.v7i4.11.20689.

[20] A. B. Sallow, Z. M. Taha, and A. S. Nori, "An Investigation for Steganography using Different Color System," *AL-Rafidain Journal of Computer Sciences and Mathematics*, vol. 7, no. 3, pp. 91–108, 2010.

[21] A. B. Sallow, M. Abdlqader, N. E. Tawfiq, and M. A. Shallal, "Initiating an Outcome-Based Education Environment at a Higher Education Institution: A Case Study," *Academic Journal of Nawroz University*, vol. 8, no. 3, pp. 39–49, 2019.

[22] A. B. Sallow, "Design And Implementation Distributed System Using Java-RMI Middleware," *Academic Journal of Nawroz University*, vol. 9, no. 1, pp. 113–120, 2020.

[23] H. K. Shaikha and A. B. Sallow, "Mobile Cloud Computing: A Review," *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 129–134, 2017.

[24] D. B. Abdullah and A. B. Sallow, "EOE-DRTSA: End-to-End Distributed Real-time System Scheduling Algorithm," *International Journal of Computer Science Issues*, vol. 10, no. 2, p. 8, 2013.

[25] D. B. Abdullah and A. B. Sallow, "Developing Fault Tolerance Integrity Protocol for Distributed Real Time Systems," *AL-Rafidain Journal of Computer Sciences and Mathematics*, vol. 10, no. 1, pp. 187–193, 2013.

[26] A. B. Sallow, A. A. H. Alkurdi, and Z. A. Sulaiman, "Proposed System for Educational Augmented Reality Smart Book," *ACAD J NAWROZ UNIV*, vol. 8, no. 3, p. 61, Aug. 2019, doi: 10.25007/ajnu.v8n3a397.

[27] A. B. Sallow and Y. M. Younis, "Augmented Reality: A Review," *ACAD J NAWROZ UNIV*, vol. 8, no. 3, p. 76, Aug. 2019, doi: 10.25007/ajnu.v8n3a399.

[28] A. B. Sallow, "Android Multi-threading Program Execution on single and multi-core CPUs with Matrix multiplication," *International Journal of Engineering*, vol. 7, no. 4, p. 6, 2018, doi: 10.14419/ijet.v7i4.29340.

[29] A. Sallow and D. Abdullah, "Constructing Sierpinski Gasket Using GPUs Arrays," *International Journal of Computer Science Issues (IJCSI)*, vol. 11, no. 6, p. 131, 2014.

[30] A. B. Sallow and S. R. Hussain, "Multi-Agent System for Supporting and Managing Real Estate Marketing," *Academic Journal of Nawroz University*, vol. 9, no. 3, pp. 54–62, 2020, doi: 10.25007/ajnu.v9n3a703.

[31] A. B. Sallow and H. Kh. Shaikha, "Optical Disc and Blood Vessel Segmentation in Retinal Fundus Images," *Academic Journal of Nawroz University (AJNU)*, vol. 8, no. 3, pp. 67–75, 2019, doi: 10.25007/ajnu.v8n3a398.

[32] H. K. Shaikha and A. B. Sallow, "Optic Disc Detection and Segmentation in Retinal Fundus Image," 2019, pp. 23–28.

[33] F. A. ZULKIFLE, R. HASSAN, R. M. OTHMAN, and A. B. SALLOW, "SUPERVISED CLASSIFICATION AND IMPROVED FILTERING METHOD FOR SHORELINE DETECTION," *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 20, pp. 5628–5636, 2017.

[34] A. Sallow, *A Fault Tolerance Distributed Real-Time System. Design and Implementation*. GRIN Verlag, 2014.

[35] M. Y. Kashmola and A. B. Sallow, "Information Hiding Techniques Using Network Protocols," *AL-Rafidain Journal of Computer Sciences and Mathematics*, vol. 8, no. 2, pp. 35–50, 2011, doi: 10.33899/csmj.2011.163650.

[36] Z. Ageed, M. R. Mahmood, M. M. Sadeeq, M. B. Abdulrazzaq, and H. Dino, "Cloud Computing Resources Impacts on Heavy-Load Parallel Processing Approaches."

[37] L. M. Haji, O. M. Ahmad, S. R. Zeebaree, H. I. Dino, R. R. Zebari, H. M. Shukur, "Impact of Cloud Computing and Internet of Things on the Future Internet." *Technology Reports of Kansai University*, vol. 62, no. 5, pp. 2179–2190, Jun. 2020.

[38] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, Art. no. 2, May 2020, doi: 10.38094/jastt1224.

[39] S. R. Zeebaree, K. Jacksi, and R. R. Zebari, "Impact analysis of SYN flood DDoS attack on HAProxy and NLB cluster-based web servers," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 1, pp. 510–517, 2020.

[40] O. M. Ahmed and W. M. Abduallah, "A Review on Recent Steganography Techniques in Cloud Computing," *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 106–111, 2017.

[41] O. Ahmed and A. Brifcani, "Gene Expression Classification Based on Deep Learning," in *2019 4th Scientific International Conference Najaf (SICN)*, Apr. 2019, pp. 145–149, doi: 10.1109/SICN47020.2019.9019357.

[42] O. Alzakholi, L. Haji, H. Shukur, R. Zebari, S. Abas, and M. Sadeeq, "Comparison Among Cloud Technologies and Cloud Performance," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, Art. no. 2, Apr. 2020, doi: 10.38094/jastt1219.

[43] S. R. Zeebaree, R. R. Zebari, K. Jacksi, and D. A. Hasan, "Security Approaches For Integrated Enterprise Systems Performance: A Review," *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, vol. 8, no. 12, Dec. 2019.

[44] O. H. Jader, S. R. Zeebaree, and R. R. Zebari, "A State Of Art Survey For Web Server Performance Measurement And Load Balancing Mechanisms," *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, vol. 8, no. 12, pp. 535–543, Dec. 2019.

[45] S. R. M. Zeebaree, H. M. Shukur, L. M. Haji, R. R. Zebari, K. Jacksi, and S. M.Abas, "Characteristics and Analysis of Hadoop Distributed Systems," *Technology Reports of Kansai University*, vol. 62, no. 4, pp. 1555–1564, Apr. 2020.

[46] S. R. M. Zeebaree, L. M. Haji, I. Rashid, R. R. Zebari, O. M. Ahmed, K. Jacksi, H. M Shukur, "Multicomputer Multicore System Influence on Maximum Multi-Processes Execution Time," *TEST Engineering & Management*, vol. 83, no. May/June, pp. 14921–14931, May 2020.

[47] S. R. M. Zeebaree, B. w. salim, R. R. Zebari, H. M. Shukur, A. S. Abdulraheem, A. I. Abdulla, S. M. Mohammed, "Enterprise Resource Planning Systems and Challenges," *Technology Reports of Kansai University*, vol. 62, no. 4, pp. 1885–1894, Apr. 2020.

[48] L. Haji, R. R. Zebari, S. R. M. Zeebaree, W. M. Abduallah, H. M. Shukur, and O. Ahmed, "GPUs Impact on Parallel Shared Memory Systems Performance," *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 08, pp. 8030–8038, 21, May, doi: 10.37200/IJPR/V2418/PR280814.

[49] H. I. Dino, M. B. Abdulrazzaq, S. R. M. Zeebaree, A. B. Sallow, R. R. Zebari, H. M. Shukur, L. M. Haji, "Facial Expression Recognition based on Hybrid Feature Extraction Techniques with Different Classifiers," *TEST Engineering & Management*, vol. 83, pp. 22319–22329, 2020.

[50] L. M. Haji, S. R. Zeebaree, O. M. Ahmed, A. B. Sallow, K. Jacksi, and R. R. Zeabri, "Dynamic Resource Allocation for Distributed Systems and Cloud Computing," *TEST Engineering & Management*, vol. 83, no. May/June 2020, pp. 22417–22426, 2020.

[51] H. Shukur, S. Zeebaree, R. Zebari, D. Zeebaree, O. Ahmed, and A. Salih, "Cloud Computing Virtualization of Resources Allocation for Distributed Systems," *Journal of Applied Science and Technology Trends*, vol. 1, no. 3, pp. 98–105, 2020.

[52] H. Shukur, S. Zeebaree, R. Zebari, O. Ahmed, L. Haji, and D. Abdulqader, "Cache Coherence Protocols in Distributed Systems," *Journal of Applied Science and Technology Trends*, vol. 1, no. 3, pp. 92–97, 2020.

[53] H. I. Dino, S. R. M. Zeebaree, A. A. Salih, R. R. Zebari, Z. S. Ageed, H. M. Shukur, L. M. Haji, S. S. Hasan "Impact of Process Execution and Physical Memory-Spaces on OS Performance." *Technology Reports of Kansai University*, vol. 62, no. 4, pp. 1885–1894, Apr. 2020.

[54] H. I. Dino, S. R. Zeebaree, O. M. Ahmad, H. M. Shukur, R. R. Zebari, and L. M. Haji, "Impact of Load Sharing on Performance of Distributed Systems Computations." *International Journal of Multidisciplinary Research and Publications, vol.3. no. 1, pp.* 30-37, 2020.

[55] H. Fereidooni, M. Conti, D. Yao, and A. Sperduti, "ANASTASIA: ANdroid mAlware detection using STatic analySIs of Applications," in *2016 8th IFIP international conference on new technologies, mobility and security (NTMS)*, 2016, pp. 1–5.

[56] S. R. Zeebaree, H. M. Shukur, and B. K. Hussan, "Human resource management systems for enterprise organizations: A review," *Periodicals of Engineering and Natural Sciences*, vol. 7, no. 2, pp. 660–669, 2019.

[57] O. F. Mohammad, M. S. M. Rahim, S. R. M. Zeebaree, and F. Y. Ahmed, "A survey and analysis of the image encryption methods," *International Journal of Applied Engineering Research*, vol. 12, no. 23, pp. 13265–13280, 2017.

[58] D. Q. Zeebaree, H. Haron, A. M. Abdulazeez, and S. R. Zeebaree, "Combination of K-means clustering with Genetic Algorithm: A review," *International Journal of Applied Engineering Research*, vol. 12, no. 24, pp. 14238–14245, 2017.

[59] D. A. Zebari, H. Haron, S. R. Zeebaree, and D. Q. Zeebaree, "Multi-Level of DNA Encryption Technique Based on DNA Arithmetic and Biological Operations," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 312–317.

[60] Z. N. Rashid, S. R. Zeebaree, and A. Shengul, "Design and Analysis of Proposed Remote Controlling Distributed Parallel Computing System Over the Cloud," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 118–123.

[61] S. R. Zeebaree, A. B. Sallow, B. K. Hussan, and S. M. Ali, “Design and Simulation of High-Speed Parallel/Sequential Simplified DES Code Breaking Based on FPGA,” in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 76–81.

[62] D. A. Zebari, H. Haron, S. R. Zeebaree, and D. Q. Zeebaree, “Enhance the Mammogram Images for Both Segmentation and Feature Extraction Using Wavelet Transform,” in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 100–105.

[63] N. O. Y. Subhi R. M. Zebari, “Effects of Parallel Processing Implementation on Balanced Load-Division Depending on Distributed Memory Systems,” *J. of university of Anbar for pure science*, vol. 5, no. 3, Art. no. 3, 2011.

[64] A. M. Abdulazeez and S. R. Zeebaree, “Design and Implementation of Electronic Learning System for Duhok Polytechnic University,” *Academic Journal of Nawroz University*, vol. 7, no. 3, pp. 249–258, 2018.

[65] M. A. Ali, D. Svetinovic, Z. Aung, and S. Lukman, “Malware detection in android mobile platform using machine learning algorithms,” in *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS)*, Dubai, United Arab Emirates, Dec. 2017, pp. 763–768, doi: 10.1109/ICTUS.2017.8286109.

[66] A. M. Abdulazeez, S. R. Zeebaree, and M. A. Sadeeq, “Design and Implementation of Electronic Student Affairs System,” *Academic Journal of Nawroz University*, vol. 7, no. 3, pp. 66–73, 2018.

[67] A. S. Y. Subhi Rafeeq Mohammed Zebari, “Improved Approach for Unbalanced Load-Division Operations Implementation on Hybrid Parallel Processing Systems,” *Journal of University of Zakho*, vol. 1, no. (A) No.2, Art. no. (A) No.2, 2013.

[68] H. Han, S. Lim, K. Suh, S. Park, S. Cho, and M. Park, “Enhanced Android Malware Detection: An SVM-Based Machine Learning Approach,” in *2020 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2020, pp. 75–81.

[69] Z. N. Rashid, K. H. Sharif, and S. Zeebaree, “Client/Servers Clustering Effects on CPU Execution-Time, CPU Usage and CPU Idle Depending on Activities of Parallel-Processing-Technique Operations “,” *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, vol. 7, no. 8, pp. 106–111, 2018.

[70] B. R. Ibrahim, S. R. Zeebaree, and B. K. Hussan, “Performance Measurement for Distributed Systems using 2TA and 3TA based on OPNET Principles,” *Science Journal of University of Zakho*, vol. 7, no. 2, pp. 65–69, 2019.

[71] S. R. M. Zeebaree, K. H. Sharif, and R. M. M. Amin, “Application Layer Distributed Denial of Service Attacks Defense Technique: A Reiview,” *Academic Journal of Nawroz University (AJNU)*, vol. 7, no. 4, Art. no. 4, 2018.

[72] S. R. Zeebaree, “DES encryption and decryption algorithm implementation based on FPGA,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 2, pp. 774–781, 2020.

[73] B. S. Osanaiye *et al.*, “Network Data Analyser and Support Vector Machine for Network Intrusion Detection of Attack Type,” *REVISTA AUS*, vol. 26, no. 1, Art. no. 1, 2019.

[74] W. M. Abduallah and S. R. M. Zeebaree, “New Data hiding method based on DNA and Vigenere Autokey,” *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 83–88, 2017.

[75] I. A. Khalifa, S. R. Zeebaree, M. Ataş, and F. M. Khalifa, “Image Steganalysis in Frequency Domain Using Co-Occurrence Matrix and Bpnn,” *Science Journal of University of Zakho*, vol. 7, no. 1, pp. 27–32, 2019.

[76] “Malware analysis,” *Wikipedia*. Jun. 17, 2020, Accessed: Jun. 21, 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Malware_analysis&oldid=963081117.

[77] X. Wang, Y. Yang, and Y. Zeng, “Accurate mobile malware detection and classification in the cloud,” *SpringerPlus*, vol. 4, no. 1, p. 583, Dec. 2015, doi: 10.1186/s40064-015-1356-1.

[78] W. Park, S. Kim, and W. Ryu, “Detecting malware with similarity to Android applications,” in *2015 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju Island, South Korea, Oct. 2015, pp. 1249–1251, doi: 10.1109/ICTC.2015.7354788.

[79] L. Wen and H. Yu, “An Android malware detection system based on machine learning,” Chongqing City, China, 2017, p. 020136, doi: 10.1063/1.4992953.

[80] H. Liang, Y. Song, and D. Xiao, “An end-to-end model for Android malware detection,” in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, Beijing, China, Jul. 2017, pp. 140–142, doi: 10.1109/ISI.2017.8004891.

[81] B. Amro, “Personal Mobile Malware Guard PMMG: a mobile malware detection technique based on user’s preferences,” *Personal Mobile Malware Guard PMMG: a mobile malware detection technique based on user’s preferences*, p. 8, 2018.

[82] Y. Liu, K. Guo, X. Huang, Z. Zhou, and Y. Zhang, “Detecting Android Malwares with High-Efficient Hybrid Analyzing Methods,” *Mobile Information Systems*, vol. 2018, pp. 1–12, Mar. 2018, doi: 10.1155/2018/1649703.

[83] Y. Fan, S. Hou, Y. Zhang, Y. Ye, and M. Abdulhayoglu, “Gotcha - Sly Malware!: Scorpion A Metagraph2vec Based Malware Detection System,” in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, London United Kingdom, Jul. 2018, pp. 253–262, doi: 10.1145/3219819.3219862.

[84] S. Sen, E. Aydogan, and A. I. Aysan, “Coevolution of Mobile Malware and Anti-Malware,” *IEEE Trans.Inform.Forensic Secur.*, vol. 13, no. 10, pp. 2563–2574, Oct. 2018, doi: 10.1109/TIFS.2018.2824250.

[85] F. Shen, J. D. Vecchio, A. Mohaisen, S. Y. Ko, and L. Ziarek, “Android Malware Detection Using Complex-Flows,” *IEEE Trans. on Mobile Comput.*, vol. 18, no. 6, pp. 1231–1245, Jun. 2019, doi: 10.1109/TMC.2018.2861405.

[86] J. Pang and J. Bian, “Android Malware Detection Based on Naive Bayes,” in *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China, Oct. 2019, pp. 483–486, doi: 10.1109/ICSESS47205.2019.9040796.

[87] S. Sabhadiya, J. Barad, and J. Gheewala, “Android Malware Detection using Deep Learning,” in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, Apr. 2019, pp. 1254–1260, doi: 10.1109/ICOEI.2019.8862633.

Amira B. Sallow, et. al. “An Investigation for Mobile Malware Behavioral and Detection Techniques Based on Android Platform.” *IOSR Journal of Computer Engineering (IOSR-JCE)*, 22(4), 2020, pp. 14-20.