International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

# Proactive Response Model for App Centric Attack Recognition

**I. Gayathri[1], Ch. Vyshnavi Bhavani[2], G. Sowjanya[3], K. Lakshmi Sai[4]**

[1,2,3,4]Student, QIS College of Engineering and Technology

**Abstract**

Safeguarding strong security while reducing energy usage is still a top priority in the world of smartphone applications. In addition to addressing these important concerns, this work presents a revolutionary paradigm that advances attack modeling and intrusion detection in a proactive manner. Taking advantage of the novel framework of Application-based Behavioral Model Analysis (ABMA), our approach goes beyond traditional security protocols by deeply examining the complex behaviors of mobile applications. Through the smooth integration of advanced intrusion detection systems with ABMA, we present a comprehensive solution that maximizes energy efficiency while strengthening application security. We enable both users and developers to proactively prevent possible threats, including communication-based assaults, by rigorous analysis and modeling. This comprehensive strategy ushers in a new era of mobile security and energy efficiency by protecting sensitive data and user privacy while also ensuring an effective and sustainable ecology for smartphone applications.

**Keywords:** Applications For Smartphones, Energy Consumption, Attack Modelling, Security, Application-Based Behavioural Model Analysis(ABMA) And Intrusion Detection, Ddos Application.

## 1. INTRODUCTION

The widespread usage of mobile applications has resulted in an unsettling increase in app-centric attacks in recent times, which presents serious security risks to both individuals and companies. Our project, which we rename as the Proactive Response Model for App-centric Attack Recognition, aims to construct a behavioral model for live detections of app-based attacks in order to solve this urgent topic. Research on Android malware detection frameworks [3], [4], static analysis approaches [12], and anomaly detection methods [13] are only a few of the studies that have inspired this endeavor. Our study intends to create a strong framework that can detect and mitigate app-centric risks in real-time by combining these findings. To achieve our goals, we plan to combine different techniques and technologies to make Android devices more resistant to harmful activity. Notably, coordinated attacks meant to interfere with the functionality of apps and the availability of services will be foiled by the implementation of Distributed Denial of Service (DDoS) mitigation techniques [1], [32]. We'll also look at how well Advanced Behavioral Malware Analysis (ABMA) techniques work [2], [3] to examine app activities and spot unusual patterns that point to malicious intent. In addition, our research will make use of intrusion detection models [13, 14] to track app activity over time and identify suspect behavior quickly, allowing for early detection of new threats. Our proactive response model's detection capabilities are improved by leveraging machine learning methods, forward symbolic execution, and dynamic taint analysis [11], [16], and [15]. Using these cutting-edge methods, we hope to strengthen mobile devices' security posture by exposing hidden threats and

malicious activity within Android applications. In addition, our experiment highlights the significance of privacy-preserving methods [20]–[23] in order to protect user data and respect privacy rights when intrusive detection algorithms are being implemented. We hope that this diverse project will further app-centric security paradigms and provide a safer online environment for Android users across the globe.

## 2. PURPOSE OF THE PAPER

This work aims to provide a proactive response model for app-centric threat recognition by utilizing knowledge from previous studies and approaches [3], [4]. The main goal is to create a behavioral model for live app-based attack detection in light of the growing threat environment that targets Android devices. This model is intended to serve as a preventative measure against new app-centric threats. The goal of the paper is to outline a complete strategy for strengthening the security posture of Android ecosystems by combining findings from many research disciplines, such as Android malware detection frameworks [3], [4], static analysis approaches [12], and anomaly detection methods [13].

Additionally, the goal of the article is to investigate how different approaches and technologies might be combined to improve Android devices' resistance to harmful activity [1], [32]. Specifically, it suggests using Distributed Denial of Service (DDoS) mitigation techniques [1], [32] to counter coordinated attacks meant to interfere with the functionality of apps and the availability of services. The study also recommends using Advanced Behavioral Malware Analysis (ABMA) techniques [3], [4] to examine app actions and spot unusual patterns suggestive of malicious intent. The research aims to create a continuous monitoring system that can identify suspicious actions and enable proactive responses to emerging risks in Android environments by utilizing Intrusion Detection Models [13], [14].

Furthermore, especially in the context of Android platforms, this article attempts to address the urgent need for proactive defensive measures against constantly changing mobile threats. It is necessary to transition from reactive measures to proactive ones due to the rise in app-centric attacks and the sophistication of malware tactics [11], [16]. Through the utilization of machine learning techniques, forward symbolic execution, and dynamic taint analysis [11], [16], and [15], the study aims to improve the proactive response model's detection capabilities. By taking a proactive approach, you may protect consumers from the ever-changing world of mobile security threats by reducing the potential effect of zero-day exploits and previously unknown threats.

## 3. LITERATURE REVIEW

In their project presented at the 5th International Conference on Malicious Unwanted applications, Bläsing et al. [1] created an Android application sandbox system designed to identify suspicious applications. This novel solution preserves the integrity of the underlying device while offering a controlled environment for examining potentially hazardous Android applications. Their method improves Android device security by providing an early warning system for hazards caused by rogue apps and by separating and examining questionable software behavior.

[2] "SpyDroid," a complete framework presented in a project presented at the 13th International Conference on Malicious Unwanted Software, is presented by Iqbal et al. The framework's purpose is to make it easier to integrate several Android-based real-time malware detectors. Through the utilization of this framework, users can simultaneously deploy a wide range of malware detection technologies, improving the security posture of Android devices in general. By using a variety of detection

methodologies, SpyDroid's strategy enables users to strengthen their defenses against emerging threats, reducing the likelihood of malware infiltration and boosting the resilience of Android ecosystems.

[3] In a research that was published in the Journal of Intelligent Information Systems, Shabtai et al. unveiled "Andromaly," a behavioral malware detection framework designed exclusively for Android smartphones. Andromaly utilizes advanced behavioral analysis methods to recognize possible malware threats by analyzing their unusual behavior patterns. The framework improves the capacity to identify and mitigate malware infestations on Android devices by analyzing app actions in real-time. Andromaly greatly strengthens the security posture of Android ecosystems by taking a proactive approach to malware identification, providing consumers with a strong defense mechanism against ever-evolving threats.

[4] Dini et al. presented their project at the International Conference on Mathematical Methods, Models, Architecture, and Computer Networks Security, where they announced "MADAM," a multi-level anomaly detector for Android malware. Using a variety of traits and signs, MADAM uses a complex multi-level anomaly detection technique to find possible malware on Android devices. MADAM strengthens the security of Android ecosystems by improving malware detection accuracy and efficacy through the combination of several detection algorithms. By actively combating malware attacks, MADAM makes a substantial contribution to enhancing the robustness of Android devices and protecting users from new security dangers.

[9] In their project presented at the 15th International Conference on Network-Based Information Systems, Matsudo et al. developed a security advisory system intended to give users real-time security recommendations throughout the installation of applications on the Android operating system. The purpose of this system is to raise users' awareness of security issues by warning them about possible dangers related to installing particular applications. Through the utilization of contextual data and security indicators, the system enables users to make knowledgeable judgments about installing apps, consequently decreasing the probability of unintentionally installing harmful malware. Matsudo et al.'s suggestion enhances the security posture of Android devices and promotes a safer user experience by taking a proactive approach to security advising.

[11] According to their project presented at the Network and Distributed System Security Symposium (NDSS), Grace et al. studied the systematic identification of capability leaks in stock Android cellphones. The project's main goal is to find any security flaws in the stock Android operating system that can unintentionally allow hostile apps to access sensitive functions. Grace et al. methodically examine the Android platform in an effort to identify situations in which apps might obtain unwanted access to private information or features. They shed light on places where enhancements to the security architecture of Android smartphones can be implemented to reduce the danger of capability leaks and increase overall device security through their research.

[12] According to a project published in Frontiers of Information Technology & Electronic Engineering, Firdaus et al. suggested a novel method for Android malware detection by utilizing genetic search-based feature discovery and static analysis. Through the identification of the most pertinent characteristics that differentiate between dangerous and benign Android applications, the research seeks to improve the efficacy of malware detection. The authors methodically find and choose the ideal set of criteria for precise malware identification using a genetic search-based approach in conjunction with static analysis, which examines the code and structure of apps without executing them. This method improves the overall effectiveness of malware detection techniques by enabling more accurate classification of Android applications.

[13] Using Bayesian classification, Yerima et al. presented a unique method for Android malware detection at the IEEE 27th International Conference on Advanced Information Networking and Applications (AINA). The project's main objective is to categorize Android applications as benign or malicious using Bayesian classification techniques, taking into account a variety of features that are taken from the apps. The authors create a probabilistic model that determines the probability of a program being malicious based on its observed properties by utilizing Bayesian inference. Through the analysis and classification of app activities using statistical principles, our approach makes Android malware detection more precise and effective. Yerima et al.'s research provides a promising path forward for the development of malware detection techniques in the Android ecosystem.

[14] In their effort, Wu et al. presented "DroidMat," a unique method for Android malware detection through tracing manifest and API calls, at the 7th Asia Joint Conference on Information Security. The project's main objectives are to trace the API calls that the apps make while they are running and analyze the manifest files for Android applications. DroidMat looks for unusual activity that could point to the presence of malware by monitoring API calls and looking closely at the permissions sought in the manifest files. With this method, potentially dangerous apps can be identified by their interactions with the Android system through API calls and their access to private device resources. Wu and colleagues advance malware detection techniques with their research.

[16] At the IEEE Symposium on Security and Privacy, Schwartz et al. gave a thorough paper titled "All you ever wanted to know about dynamic taint analysis and forward symbolic execution (but might have been afraid to ask)". The project explores forward symbolic execution and dynamic taint analysis, providing a thorough understanding of their concepts, practices, and cybersecurity applications. While forward symbolic execution entails methodically investigating various program execution paths in order to find vulnerabilities or examine a program's behavior, dynamic taint analysis tracks the flow of data through a program at runtime, labeling data as "tainted" when it comes from untrusted sources. These methods offer strong instruments for identifying and evaluating security flaws and data breaches.

[17] The research presented at the 11th Annual International Conference on Mobile Systems, Applications, and Services describes how Khan et al. created "Cameo," a middleware for mobile advertisement distribution. Cameo serves as a bridge between ad networks and mobile applications, enabling the effective and focused distribution of ads to mobile consumers. The middleware utilizes an array of methodologies, including contextual analysis, user profiling, and geo-location targeting, in order to maximize the pertinence and efficacy of the adverts presented to consumers. Cameo also has privacy-preserving features to protect user information and guarantee adherence to privacy laws. The goal of Khan et al.'s research is to increase the monetization potential of mobile applications while giving users a smooth and unobtrusive advertising experience.

[21] In their presentation presented at the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS), Ev_mievski et al. explored techniques for limiting privacy breaches in privacy-preserving data mining. The project's main goal is to preserve people's privacy while still permitting insightful data analysis. Random data perturbation—adding noise to datasets to mask sensitive information—and differential privacy—which guarantees that individual data contributions do not excessively influence analysis outcomes—are two of the techniques used. Furthermore, data anonymization techniques like l-diversity and k-anonymity are suggested to suppress or generalize identifiable traits while maintaining anonymity. Collectively, these strategies seek to achieve a balance

between privacy protection issues and data analysis requirements, which aids in the development of ethical data mining techniques.

[23] "OB-PWS," an obfuscation-based private web search system, was proposed by Balsa et al. and described in depth in their paper presented at the IEEE Symposium on Security and Privacy. In order to prevent search engines from tracking and profiling users, the project obfuscates user searches, hence addressing privacy concerns in web search. In order to hide the user's search intentions while still returning pertinent search results, OB-PWS uses obfuscation techniques like query blurring and result perturbation. OB-PWS attempts to improve user privacy and reduce the possibility of tracking and surveillance by third parties by hiding search queries. This method advances privacy-preserving technology in the internet search space by enabling users to perform searches without compromising their privacy.

[31] "Towards statistical queries over distributed private user data" was the project title that Chen et al. presented at the NSDI conference. The goal of the project is to protect user privacy while enabling statistical inquiries on scattered user data. The authors suggested methods, such as safe multiparty computation protocols and differential privacy mechanisms, to solve privacy problems in distributed data analysis scenarios. Secure multiparty computation enables many parties to collaboratively compute aggregate statistics on their data without disclosing individual data points, while differential privacy guarantees that statistical inquiries do not reveal sensitive personal information. Chen et al. hope to advance the field of privacy-preserving distributed data analysis by implementing these strategies to provide collaborative data analysis while safeguarding user privacy.

[32] Dierks wrote a technical report that described version 1.2 of the Transport Layer Security (TLS) protocol. This protocol uses a variety of cryptographic approaches to guarantee the secrecy, integrity, and authenticity of data transferred between client and server applications. It is frequently used to secure communication over computer networks. TLS 1.2 includes key exchange protocols like RSA and Diffie-Hellman for secure key establishment, as well as sophisticated encryption algorithms like AES and HMAC for symmetric encryption and message authentication. Furthermore, TLS 1.2 facilitates the use of digital certificates from reliable certificate authority to confirm the identity of parties involved in communication. TLS 1.2 offers a secure framework for creating encrypted connections over the internet, protecting sensitive data from interception and manipulation with its strong cryptographic methods.

[33] The SIGCOMM Computer Communications Review publication by Chen et al. discussed "Splitx," a project centered on high-performance private analytics. By making it possible to analyze sensitive data in an effective and scalable manner while protecting individual privacy, the initiative tackles privacy concerns in data analytics. Splitx performs computations on encrypted data without disclosing the underlying material to any person engaged in the computation by utilizing methods like homomorphic encryption and secure multiparty computing (MPC). Splitx ensures that the privacy of individual data providers is preserved while enabling numerous parties to evaluate data cooperatively. This is achieved by utilizing these privacy-preserving strategies. With this method, businesses can extract insightful information from sensitive data without violating the privacy rights of individuals.

## 4. METHODOLOGY

### A. Intrusion detection model

An essential part of cybersecurity frameworks is an intrusion detection model, which protects computer networks and systems from harmful or unauthorized access. Acting as a watchdog, it closely examines user activity, system logs, and network traffic in real-time or almost real-time, looking for anomalies or

patterns that could point to security breaches. These models, which can be broadly divided into signature-based and anomaly-based detection techniques, use complex algorithms to spot known attack patterns or departures from expected behavior.

Feature=*f*(application data)

$z=b+w1x1+w2x2+\ldots+wnxn$

$\hat{}=\sigma(z)=1+e-z1$

**B. ABMA**

A cryptographic access control architecture that allows for fine-grained access control in distributed systems is called the Attribute-Based Multi-Authority (ABMA) scheme. Instead than focusing on a person's identity, ABMA defines access controls based on the properties of the user, such as roles, traits, or credentials. This methodology facilitates adaptable and dynamic access control, whereby access determinations are predicated on the fulfillment of designated attribute-based regulations. Furthermore, many independent authorities work together to manage user attributes and enforce access regulations in multi-authority situations, which are supported by ABMA. ABMA guarantees safe and privacy-preserving access control by utilizing cryptographic techniques such as attribute-based encryption (ABE). This makes it appropriate for a variety of applications, such as cloud computing, Internet of Things, and decentralized systems.

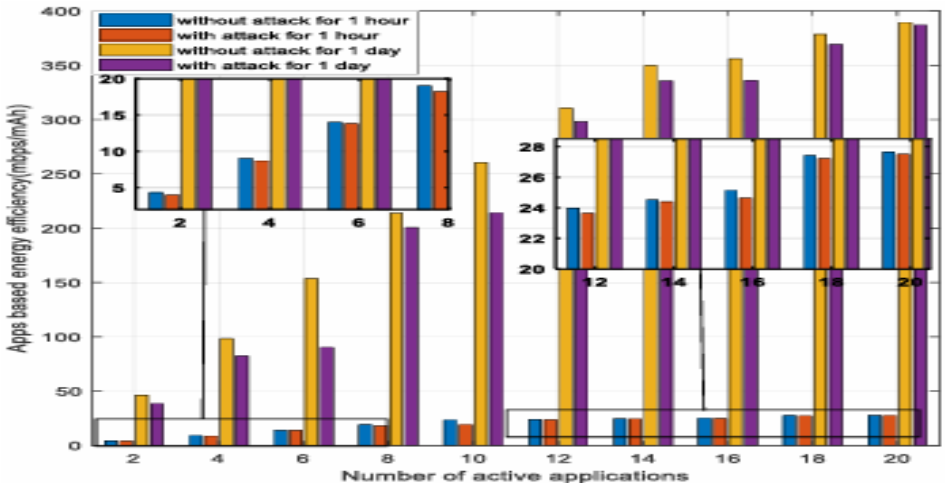



**Fig 1:Apps based energy consumption.**

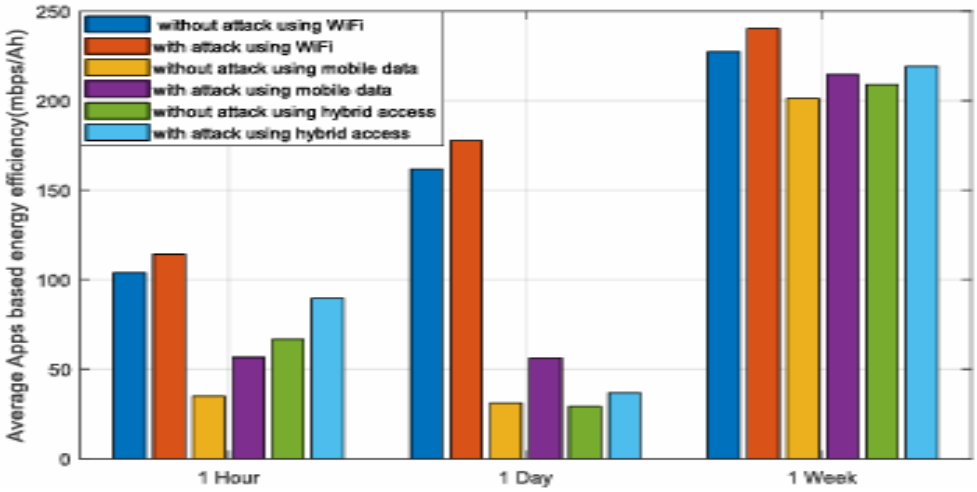



**Fig2: Apps based energy efficiency analysis**

It is seen that applications with lower ad refresh rates, such as those lasting 20 and 30 seconds, consume 3MB to 5.5MB of bandwidth over the tested duration; these apps account for approximately 70% of all the apps utilized in our tests. Thirty percent of apps use between 0.5MB and 2.5MB of bandwidth (ad refresh rates between 45 and 60 seconds). According to Figure 5's inner graph, which displays the PDF of network bandwidth consumption, 26.30% of the apps have high 4 4.5 MB bandwidth usage. Now, we assess the tested apps' computing overhead. The recorded CPU usage varies throughout apps, however not significantly: 25% to 30% of the CPU is used by CPU-intensive programs, such those in the games category, whereas less-interactive apps .

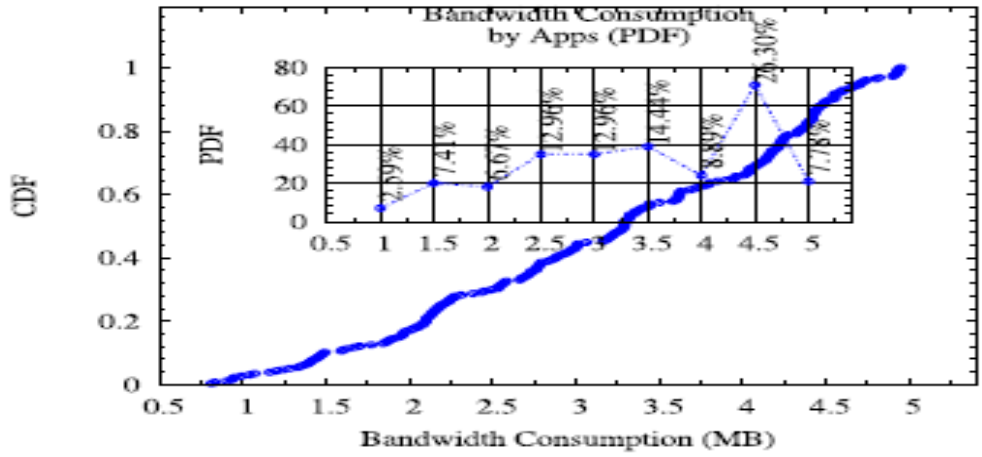



**FIGURE Bandwidth consumption (MB) calculated for ads by the apps during experiments.**

The probability density function (inner graph) and the cumulative distribution of the bandwidth utilized by apps (outer graph) are displayed in the above figure. The bandwidth usage displayed is for 2.5 hours of continuous use for each of the apps chosen, as indicated by the above figure.

**C. DDoS**

An attempt is made maliciously to stop a specific server, service, or network from operating normally by flooding it with a large amount of traffic from various sources on the internet through a Distributed Denial of Service (DDoS) attack. A denial-of-service (DDoS) assault involves the attacker taking control of a large number of compromised devices, often known as a botnet, and instructing them to send a huge number of requests or data packets to the target at the same time. The target's bandwidth, processing power, or memory are depleted by this onslaught of inbound traffic, making it unavailable to authorized users. DDoS attacks have the potential to seriously harm a business or service's reputation in addition to causing downtime and large financial losses.

## 5. RESULTS AND DISCUSSIONS

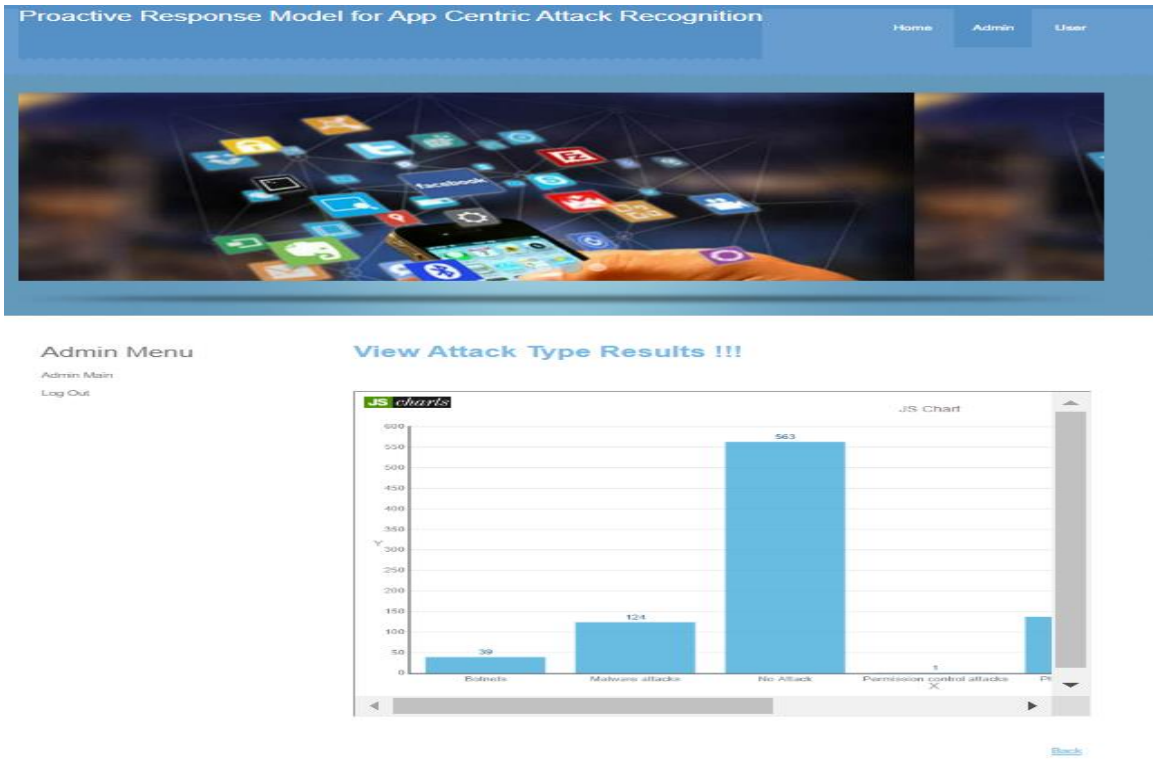
The desire of consumers to protect their privacy greatly influences whether or not they utilize privacy apps. There has been a significant surge in privacy awareness in the public and regulatory spheres in recent decades. The revelation of widespread monitoring practices [31] and inadvertent disclosures of datasets containing private information have served as the driving forces behind this. As a result, more people are interested in implementing bespoke or personal privacy technologies.

**Fig1: Admin page**


**Fig2: Apps attack types.**

**Fig3: DDoS Attack**


**Fig4: Attack Type result**



IoT and mobile devices are booming, and their uses are commonplace globally. Meanwhile, the Industry 4.0 trend poses a threat since it blends industrial control systems with the online world. For regular work, mobile and IoT applications offer easy connectivity to the emerging network environment. Malicious mobile applications have the potential to hack mobile devices and result in financial losses since mobile phones carry private information. Additionally, in the emerging network environment, attackers could use denial-of-service assaults (DDoS) to deplete a resource of mobile or IoT devices.

## 6. CONCLUSION

Through the use of mobile applications, which allow users to access or operate IoT devices remotely through their smartphones in an emerging network environment, IoT devices and smartphones have been integrated. Malicious mobile apps have the potential to breach network security, compromise IoT devices, and steal private data. Thus, in order to prevent intrusions, mobile application security needs to be guaranteed. This paper suggests a hierarchical data model and static taint analysis-based power-efficient mobile malware detection technique. The results of the experiment demonstrate that the suggested strategy is capable of successfully identifying both malicious and benign applications. It is appropriate for large-scale malware detection or anti-virus software deployed on smartphones or Internet of Things devices because it uses less power and processing resources than dynamic analysis. The experimental results show that, because the suggested method is based on misbehavior patterns rather than signatures, it can efficiently identify new malware and malware variants whereas commercial anti-virus software is unable to do so. The adoption of 5G and IoT networks in the future will increase the frequency of mobile attacks. To better defend against varied attacks across various heterogeneous networks, an automated threat pattern creation is required.

## REFERENCES

1. T. Bläsing, L. Batyuk, A.-D. Schmidt, S. A. Camtepe, and S. Albayrak, ``An Android application sandbox system for suspicious software detection," in *Proc. 5th Int. Conf. Malicious Unwanted Softw.*, Oct. 2010, pp. 55_62.
2. S. Iqbal and M. Zulkernine, ``SpyDroid: A framework for employing multiple real-time malware detectors on Android," in *Proc. 13th Int. Conf. Malicious Unwanted Softw.*, 2018, pp. 1_8.
3. A. Shabtai, U. Kanonov,Y. Elovici, C. Glezer, andY.Weiss, ```Andromaly': A behavioral malware detection framework for Android devices," *J. Intell. Inf. Syst.*, vol. 38, no. 1, pp. 161_190, 2012.
4. G. Dini, F. Martinelli, A. Saracino, and D. Sgandurra, ``MADAM: A multi-level anomaly detector for Android malware," in *Proc. Int. Conf. Math. Methods, Models, Archit. Comput. Netw. Secur.* Cham, Switzerland: Springer, 2012, pp. 240_253.
5. G. Claudiu. *Obfuscapk*. Accessed: Jul. 25, 2020. [Online]. Available: https://github.com/ClaudiuGeorgiu/Obfuscapk
6. SRILAB. *Deguard*. Accessed: Jul. 25, 2020. [Online]. Available: https://eth-sri.github.io/deguard
7. *Gyoonus*. Accessed: Jul. 25, 2020. [Online]. Available: https://github. com/Gyoonus/deoptfuscator
8. F. Di Cerbo, A. Girardello, F. Michahelles, and S. Voronkova, ``Detection of malicious applications on Android OS," in *Proc. Int.Workshop Comput. Forensics*. Cham, Switzerland: Springer, 2010, pp. 138_149.

9. T. Matsudo, E. Kodama, J. Wang, and T. Takata, ``A proposal of security advisory system at the time of the installation of applications on Android OS," in *Proc. 15th Int. Conf. Network-Based Inf. Syst.*, Sep. 2012, pp. 261_267.
10. A. Feizollah, N. B. Anuar, R. Salleh, G. Suarez-Tangil, and S. Furnell, ``AndroDialysis: Analysis of Android intent effectiveness in malware detection," *Comput. Secur.*, vol. 65, pp. 121_134, Mar. 2017.
11. M. C. Grace, Y. Zhou, Z. Wang, and X. Jiang, ``Systematic detection of capability leaks in stock Android smartphones," in *Proc. NDSS*, vol. 14, 2012, p. 19.
12. A. Firdaus, N. B. Anuar, A. Karim, and M. F. A. Razak, ``Discovering optimal features using static analysis and a genetic search based method for Android malware detection," *Frontiers Inf. Technol. Electron. Eng.*,vol. 19, no. 6, pp. 712_736, Jun. 2018.
13. S. Y. Yerima, S. Sezer, G. McWilliams, and I. Muttik, ``A new Android malware detection approach using Bayesian classi_cation," in *Proc. IEEE 27th Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, Mar. 2013, pp. 121_128.
14. D.-J. Wu, C.-H. Mao, T.-E. Wei, H.-M. Lee, and K.-P. Wu, ``DroidMat: Android malware detection through manifest and API calls tracing," in *Proc. 7th Asia Joint Conf. Inf. Secur.*, Aug. 2012, pp. 62_69.
15. N. Milosevic, A. Dehghantanha, and K.-K. R. Choo, ``Machine learning aided malware classi_cation of Android applications," *Comput. Electr. Eng.*, vol. 61, pp. 266_274, 2017.
16. E. J. Schwartz, T. Avgerinos, and D. Brumley, ``All you ever wanted to know about dynamic taint analysis and forward symbolic execution (but might have been afraid to ask)," in *Proc. IEEE Symp. Secur. Privacy*, May 2010, pp. 317_331.
17. A. J. Khan, K. Jayarajah, D. Han, A. Misra, R. Balan, and S. Seshan, ``Cameo: A middleware for mobile advertisement delivery," in *Proc. 11th Annu. Int. Conf. Mobile Syst., Appl., Services*, 2013, pp. 125_138.
18. H. Haddadi, P. Hui, and I. Brown, ``MobiAd: Private and scalable mobile advertising," in *Proc. 5th ACM Int. Workshop Mobility Evolving Internet Archit. (MobiArch)*, 2010, pp. 33_38.
19. R. Dingledine, N. Mathewson, and P. Syverson, ``Tor: The secondgeneration onion router," Tech. Rep., 2004.
20. R. Agrawal and R. Srikant, ``Privacy-preserving data mining," *ACM SIG- MOD Rec.*, vol. 29, no. 2, pp. 439_450, 2000.
21. A. Ev_mievski, J. Gehrke, and R. Srikant, ``Limiting privacy breaches in privacy preserving data mining," in *Proc. 22nd ACM SIGMOD- SIGACT-SIGART Symp. Princ. Database Syst. (PODS)*, 2003, pp. 211_222.
22. H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, ``On the privacy preserving properties of random data perturbation techniques," in *Proc. 3rd IEEE Int. Conf. Data Mining*, Nov. 2003, pp. 99_106.
23. E. Balsa, C. Troncoso, and C. Diaz, ``OB-PWS: Obfuscation-based private Web search," in *Proc. IEEE Symp. Secur. Privacy*, May 2012, pp. 491_505.
24. N. Mor, O. Riva, S. Nath, and J.Kubiatowicz, ``Bloom cookies:Web search personalization without user tracking," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2015, pp. 1_15.
25. B. H. Bloom, ``Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422_426, Jul. 1970.
26. P. Samarati and L. Sweeney, ``Generalizing data to provide anonymity when disclosing information (abstract)," in *Proc. 17th ACM SIGACT- SIGMOD-SIGART Symp. Princ. Database Syst. (PODS)*, vol. 98, 1998, p. 188.

27. L. Sweeney, ``K-anonymity: A model for protecting privacy,'' *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557_570, Oct. 2002.
28. T. Chen, A. Chaabane, P.-U. Tournoux, M. Kaafar, and R. Boreli, ``How much is too much? Leveraging ads audience estimation to evaluate public profile uniqueness,'' in *Proc. Privacy Enhancing Technol. Symp. (PETS)*, 2013, pp. 225_244.
29. A. Narayanan and V. Shmatikov, ``Robust de-anonymization of large sparse datasets,'' in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2008, pp. 111_125.
30. D. Cynthia, ``Differential privacy,'' in *Proc. 33rd Int. Colloq. Automata, Lang. Program. (ICALP)*, 2006, pp. 1_12.
31. R. Chen, A. Reznichenko, P. Francis, and J. Gehrke, ``Towards statistical queries over distributed private user data,'' in *Proc. NSDI*, vol. 12, 2012, p. 13.
32. T. Dierks, ``The transport layer security (TLS) protocol version 1.2,'' Tech. Rep., 2008.
33. R. Chen, I. E. Akkus, and P. Francis, ``Splitx: High-performance private analytics,'' *SIGCOMM Comput. Commun. Rev.*, vol. 43, pp. 315_326, Aug. 2013.