SECURITY ANALYSIS TESTING ON INDONESIAN GOVERNMET ANDROID APPLICATION USING REPACKAGING ATTACK

A THESIS

A Partial Requirement to Fulfill for master's degree in computer science

ALFIAN DODY FIRNANDO

NIM: 202110130031

Supervisor

Prof. Ir. Teddy Mantoro, M.Sc., PhD

Umar Aditiawarman, S.T., M.Sc., PhD

SCHOOL OF COMPUTER SCIENCE

NUSA PUTRA UNIVERSITY

2024

STATEMENT OF AUTHENTICITY

The undersigned below

Name	:	Alfian Dody F
ID of Student	:	202110130031
Faculty	:	Computer of Science
The Title of Thesis	:	Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack

Hereby declares truthfully

1. The work and writing of this thesis were carried out by myself without the use of any unauthorized assistance.
2. This thesis has not been submitted to any other educational institution for the award of any academic degree or certification

If proven to violate any of the above points, I am willing to accept the sanctions given by the faculty leader.

Sukabumi, 12th January 2024

Writer,

Materai 10000

Alfian Dody F
NIM: 202110130031

APPROVAL OF THESIS

Title	:	Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack
Name	:	Alfian Dody F
ID of Student	:	202110130031

Theis thesis has been review and approved

Sukabumi, 12 January 2024

Head of Study Program	Supervisor
Prof. Ir. Teddy Mantoro, M.Sc., PhD	Prof. Ir. Teddy Mantoro, M.Sc., PhD
NIDN.	NIDN.

THESIS APPROVAL

Title	:	Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack
Name	:	Alfian Dody F
ID of Student	:	202110130031

This thesis has been examined and defended before the board of examiners during the thesis session on January, 12 2024. Upon our assessment, we find this thesis to be satisfactory in quality for the awarding of the Master of Computer Science degree.

Sukabumi, 12 January 2024

Supervisor 1,

Prof. Ir. Teddy Mantoro, M.Sc., PhD
NIDN.

Examiner 1,

Dini Oktarina Dwi Handayani, S.T., M.Sc
NIDN.

Supervisor 2,

Umar Aditiawarman, S.T., M.Sc., PhD
NIDN.

Examiner 2,

Jelita Asian, M.Sc., PhD
NIDN.

PUBLICATION APPROVAL

As a member of the academic community of Nusa Putra University, i undersigned

Title	:	Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack
Name	:	Alfian Dody F
ID of Student	:	202110130031
Type of Work	:	Thesis

In the interest of scientific progress, I hereby agree to grant the University of Nusa Putra a Non-Exclusive Royalty-Free Right to my scientific work titled: "Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack," along with any necessary supporting devices (if needed). With this non-exclusive royalty-free right, the University of Nusa Putra is entitled to store, transfer media/formats, process into a database, maintain, and publish my thesis, provided that I am maintained as the author/creator and retain copyright ownership. I make this statement truthfully.

Made in Sukabumi

At the Date of: 12 January 2024

That Sates

(Alfian Dody F)

ABSTRACT

SECURITY ANALYSIS TESTING ON INDONESIAN GOVERNMENT ANDROID APPLICATION USING REPACKAGING ATTACK

The increasing adoption of digital technology by the Indonesian government has encouraged the development of a number of Android applications to make services easier for the public. However, at the same time, Android applications currently being developed have a very high vulnerability to reverse engineering, where this vulnerability can be easily exploited using a technique called a repackaging attack. Since Android was created in 2008, the threat of repackaging attacks continues to haunt Android users, including users of Indonesian government applications. Apart from that, the security integrity of the Indonesian government's Android application cannot be known for certain, especially the security threats related to repackaging attacks. This is likely because research that comprehensively evaluates the level of resistance of Indonesian government Android applications to repackaging attacks is still limited. This research develops the Metasploit payload injection technique as a repackaging method that can be used to evaluate the level of resilience of Indonesian government applications. From the results of testing 200 applications, it was found that 191 Indonesian government Android applications could still be easily exploited using repackaging attack techniques. In other words, Indonesian government applications have high potential to be used as a medium for spreading malware. This research also offers strategic recommendations for improving the integrity and security of Indonesian government applications against repackaging attacks, with the hope of protecting user data and privacy and increasing public trust in government digital applications and services.

Keyword: Repackaging Attack, Android Apps, Reverse Engineering, Android Security, Malware

FOREWORD

All praise to Allah SWT, who has given His grace and guidance so that the researchers can complete their thesis titled "Security Analysis Testing on Indonesian Government Android Application Using Repackaging Attack".

This thesis was prepared as an invitation to obtain a master's degree at Nusa Putra University, Sukabumi. The success of the author in completing this thesis was not merely the author's effort, but also thanks to the direction, support, and prayers of various parties. For that, the author wishes to convey his sincere thanks to:

1. Families, especially parents, wives and children, who tirelessly support both spiritually and materially.
2. Prof. Ir. Teddy Mantoro, M.Sc., PhD as the first lecturer and tutor who has provided guidance, direction, and motivation during the writing process of this thesis
3. Umar Aditiawarman, S.T., M.SC., PhD as the lecturer, tutor, and first mentor who has given guidance and motivations during this writing process
4. All Masters of Computer Science Lecturers who have provided very useful knowledge during lectures
5. Colleagues from the master of computer science of the University of Nusa Putra who have given support and Motivation
6. Other parties that we can not mention one by one, but have also provided support in the process of writing the thesis

The author realizes that there are still many shortcomings in the writing of this thesis. Therefore, we look forward to criticism, input, and suggestions of a constructive nature that can help to complete this thesis. Furthermore, the author hopes that the results of this research can be useful or contribute to science in the field of information security systems and society in general.

Sukabumi,   January 2024

Penulis

LIST OF CONTENTS

LIST OF TABLE

LIST OF FIGURES

APPENDIX LIST

LIST OF ABBREVIATIONS

Abbreviations	**Name**
APK	Android Package
Malware	Malicious software
ART	Android Runtime
HAL	Hardware Abstraction Layer
DEX	Dalvic Executable
Meterpreter	Metasploit Interpreter
SAI	Split APK Installer
MobSF	Mobile Security Framework
AAPT	Android Asset Packaging Tool
RASP	Runtime Application Self Protection
NRP	Native Repackaging Protection

CHAPTER I
INTRODUCTION

1.1 Research Background

Since its release by Google in September 2008, Android has become the most widely used mobile operating system in various parts of the world (Yang et al., 2022). Based on data released by StatCounter, in February 2024, Android's market share was the highest of any other mobile operating system, at 71.43 percent (Statcounter Global Stats, 2024). The main factor that makes Android have the most users in the world is its open source development, which makes it easy to adapt and modify and offers extensive flexibility and customization (Sihag et al., 2021). Android also has hardware support and an extensive application ecosystem (Zhu et al., 2023). As of January 2024, the number of Android applications available on the Google Play Store was 2,423,984 (AppBrain, 2024). This success has led to Android being installed on various devices, ranging from smartphones, tablets, smartwatches, and Internet of Things (IoT) devices. This condition makes Android have an influence on every aspect of life in the world, including Indonesia.

Smartphones with the Android operating system also have a large and significant market share in Indonesia. According to data released by StatCounter, in February 2024, Android dominated the mobile operating system market share, reaching 89.02 percent (StatCounter Global Stats, 2024). Along with the massive adoption of the Android system, the Indonesian government has launched various applications to improve public access to government services, ranging from licensing, tax payments, the application of population documents, and public health services. Based on Google Play Store data, Indonesian government apps are also widely used by the public. For example, the SATUSEHAT Mobile application managed by the Ministry of Health has been downloaded by more than 50 million people (Google Play, 2023b). In terms of public services, the MyTelkomsel app, managed by the Ministry of SOEs, has been downloaded by more than 100 million people(Google Play, 2023a).

However, the popularity of Android and its open development methods have posed significant security challenges. Globally, security threats to Android continue to occur. According to a report by Kaspersky Lab, more than 1.6 million pieces of malware had been installed on Android devices as of December 2022 (AO Kaspersky Lab, 2023). This trend is expected to continue as Android users grow and the complexity of cyberattacks increases. One of the major security issues faced by Android is repackaging attacks (Garg & Baliyan, 2021). Application repackaging poses a significant and widespread danger to the Android ecosystem (He et al., 2019). In this attack, attackers reverse engineer the original app, modify the code, embed malware or malicious code, and then repackage it (Ohello, 2023). When repackaged apps are distributed, downloaded, and installed by users, it can allow attackers to steal personal information, monitor activity, and take control of users' devices.

Repackaging attacks are the main vector for malware spread (Vidas & Christin, 2013). According to a previous study, 86 percent of the spread of malware in the Android ecosystem is done through repackaging attack techniques (Rizvi et al., 2019). The main factor that makes Android applications vulnerable to repackaging attacks is that the basic code base of Android applications is Java and Kotlin, making it easy to be compiled and re-modified (Martín & Hernández, 2019). The reverse engineering process is clarified and made easier with many tools that are freely available on the Internet, ranging from apktool, jadx (Skylot, 2020), jd-gui (Emmanuel Dupuy, 2019), dex2jar (Bob Pan, 2015), and others. Until now, almost all types of Android applications have been vulnerable to repackaging attacks. This is because Android app development pays little attention to security. According to previous research (Wermke et al., 2018), 75 percent of Android apps in the Google Play Store are not protected, making them vulnerable to repackaging attacks. The inherent characteristic of the Android operating system, which allows users to install apps from unverified sources, contributes to the heightened susceptibility to repackaging attacks (Karthick & Binu, 2017).

Repackaging attacks also threaten Android apps in the government sector. According to a report by Promon in the fourth quarter of 2022, government Android apps in some countries are vulnerable to repackaging (Promon, 2022). According to Promon, in Brazil, four out of six apps could be repackaged, and in the European Union, four out of five apps could be repackaged (Promon, 2022). Considering issues related to the vulnerability of Android apps from repackaging attacks and reports released by Promon, it can be hypothesized that Android apps developed by the Indonesian government are also vulnerable to repackaging attacks. This assumption is reinforced by the initial test results of the JKN Mobile app. The Android app developed by “BPJS Kesehatan” has a vulnerability to repackaging attacks, in which case it can be easily modified and injected with Metasploit payload code. Using the msfvenom feature of the Metasplait Framework, the repackaging attack on JKN's Mobile application only takes less than two minutes

Although this threat is clear and alarming, research on repackaging attacks against Indonesian government Android apps is still limited or to put it another way, has not been explored or analyzed in depth. In addition, the current method of repackaging attacks, namely by inserting the Metasploit payload, cannot provide an accurate picture of the level of resilience of Indonesian government Android applications. Due to the urgency and significance of the issue at hand, a security analysis of the Indonesian government's applications is essential. The researchers behind this thesis hope to be able to use it to pinpoint potential vulnerabilities in the Indonesian government's Android apps, particularly those vulnerable to repackaging attacks. Researchers hope their findings will help guide Indonesian authorities and app developers toward making government apps for Android more secure.

1.2 Problem Statement

Along with the massive use of mobile devices in Indonesia, Android apps have become an important tool for governments in providing public services and information to the public. The Indonesian government has developed many Android applications in various government sectors, ranging from healthcare, education, to applications that handle sensitive data and information. However, at the same time, Android applications currently being developed have a very high vulnerability to reverse engineering, where this vulnerability can be easily exploited using a technique called a repackaging attack. When a repackaging attack is successfully executed, one of the potential negative impacts is the leakage of users' personal data

Moreover, although repackaging attacks are widely known within the cybersecurity community and a lot of research and security services have proposed techniques to anticipate repackaging attacks, most Android apps circulating in the Google Play Store still fail to mitigate the attack. This means that these applications are still easy to repackage. Referring to this condition, Android applications developed by the Indonesian government are also not immune to the risk of repackaging attacks, raising an important question about the extent of the resilience of Indonesian government Android applications against potential repackaging attacks.

1.3 Research Objectives

Based on the formulation of the problem and the results of the literature study that has been done, it is known that Android applications are very vulnerable to repackaging attacks. Thus, the objectives of this research can be summarized as follows:

1. Conduct an analysis, evaluation, and testing of the government-owned application in Indonesia in order to determine the extent to which the application is resistant or capable of withstanding repackaging attacks.
2. Proposing a repackaging technique that can be used to test the resilience of Android apps against repackaging attacks

1.4 Significance of Research

The benefits of this research are as follows

1. This research is expected to increase understanding of the threat of repackaging attacks and their impact on government applications.
2. The results of this research are expected to help the government and application developers improve application security from repackaging attacks.

1.5 Limitation of Problems and Assumptions

In this research, the limitation of the problem is determined to clarify the scope of the research. The problem limitations in this study are:

1. Research is limited to Android applications.
2. The malware used in this research is malware generated by the Metasploit Framework.
3. The repackaging technique used is to inject the Metasploit payload.

1.6 Thesis Structure

The systematics of research in this study are organized into five sections, which consist of:

1. Chapter I Introduction

   In the introduction, it discusses the background of the problem, the objectives of the research, problem boundaries, research methods, research systematics, and research contribution.
2. Chapter II Literature Review

   This chapter discusses supporting theories and the results of scientific research related to the problems discussed.
3. Chapter III Research Methodology

   The research design section discusses the general research method, the flow of the research process, and the processes that will be carried out in this study.
4. Chapter IV Research Results and Discussion

This chapter discusses the results of the research step by step based on the research design, from problem identification to testing and evaluation.

5. Chapter V Conclusion and Recommendations

   This chapter presents conclusions from the overall research results. This chapter also proposes suggestions for future research.

CHAPTER V

CONCLUSION AND RECOMMENDATIONS

2.1 Conclusion

The results of this study show that Android applications developed by the Indonesian government still have a very significant vulnerability to repackaging attacks. Of the 200 apps tested, the surprising finding is that the vast majority, 191 apps, show vulnerability to repackaging attacks. This indicates that 95.5% of the tested apps do not have adequate security measures in place to mitigate repackaging attacks. This finding also indicates that Android app developers in the government sector pay little attention to security aspects when developing Android apps. In addition, the very low level of resilience against repackaging attacks may also cause Indonesian government Android apps to potentially become a medium for malware distribution.

Of the nine applications that were able to detect repackaging attacks, it can be concluded that the anti-repackaging techniques embedded in these applications are very effective in anticipating repackaging attacks. Applications that have been repacked with malware immediately crash or display security warnings when run on Android devices. However, some applications that have anti-repackaging techniques can also run malware, although in the end, the functions of the malware cannot be run. Meanwhile, for Indonesian government Android applications that are still vulnerable to this type of attack, application developers need to immediately implement anti-repackaging mechanisms. This is because the impact of repackaging attacks is quite significant.

In addition, the finding of a very high vulnerability also shows that the method used in the study is quite effective in testing or evaluating the resilience of Indonesian government Android applications to repackaging attacks. This is at least evident from the results of the comparison with msfvenom, where, from testing using 10 Indonesian government Android applications, the research method used in this study displays more accurate results to describe the status of the resilience level of Android applications to repackaging attacks.

2.2 Recommendation

Although the repackaging attack method used in this study is quite effective in testing or evaluating the level of resilience of Indonesian government Android applications against repackaging attacks. This research still has limitations that can be developed further in the future. The limitations that need attention are

1. The repackaging attack process is still done manually, so this method will not be efficient if used to test a large number of application samples.
2. The Metasploit malware used in this research is still detected by anti-virus and Google Play Protect.

Responding to some of the limitations above, further research can develop a framework that can automate these stages. In addition, it is also necessary to implement stages so that embedded malware can avoid detection of the Android security system, one of which is by implementing malware obfuscation. For example, by adopting the obfuscation technique as done by Aonzo et al (2020)

In relation to Indonesian government Android applications that are still very vulnerable to repackaging attacks, application developers need to implement anti-repackaging techniques. From the literature study that has been done, there are several studies that can be used as a reference for application developers to anticipate repackaging attacks, namely AppIS (Song et al., 2018), Self Defending Code (Chen et al., 2018), Bombdroid (Zeng et al., 2018), CodeCloak (He et al., 2019), NRP (He et al., 2019), and ARMAND (Merlo et al., 2021a). In addition to referring to some of these studies, Android application developers in the government sector can also utilize mobile application security services such as DexGuard, Verimatrix, AppDome, Promon Shield, and so on. Application developers can also utilize freeRASP (Talsec, n.d.) technology, which is open source and can be obtained for free.

Meanwhile, Indonesian government Android app developers who have implemented anti-repackaging techniques should also consider using multi-pattern anti-repackaging protection. For example, by combining signature verification techniques with RASP (Runtime Application Self Protection), it is expected that the anti-repackaging technique is not easily detected and disabled by attackers.

REFERENCE

AO Kaspersky Lab. (2023). *200,000 new mobile banking Trojan installers discovered, double the 2021*. https://www.kaspersky.com/about/press-releases/2023_200000-new-mobile-banking-trojan-installers-discovered-double-the-2021

Aonzo, S., Georgiu, G. C., Verderame, L., & Merlo, A. (2020). Obfuscapk: An open-source black-box obfuscation tool for Android apps. *SoftwareX*, *11*, 100403. https://doi.org/10.1016/j.softx.2020.100403

AppBrain. (2024, July 23). *Number of Android apps on Google Play*. https://www.appbrain.com/stats/number-of-android-apps

Berthome, P., Fecherolle, T., Guilloteau, N., & Lalande, J.-F. (2012). Repackaging Android Applications for Auditing Access to Private Data. *2012 Seventh International Conference on Availability, Reliability and Security*, 388–396. https://doi.org/10.1109/ARES.2012.74

Bob Pan. (2015). *GitHub - pxb1988/dex2jar: Tools to work with android .dex and java .class files*. https://github.com/pxb1988/dex2jar

Bostani, H., & Moonsamy, V. (2024). EvadeDroid: A practical evasion attack on machine learning for black-box Android malware detection. *Computers & Security*, *139*, 103676. https://doi.org/10.1016/j.cose.2023.103676

Chen, K., Zhang, Y., & Liu, P. (2018). Leveraging Information Asymmetry to Transform Android Apps into Self-Defending Code Against Repackaging Attacks. *IEEE Transactions on Mobile Computing*, *17*(8), 1879–1893. https://doi.org/10.1109/TMC.2017.2782249

de la Torre-Diez, I., Trinchet, B. O., Rodrigues, J. J. P. C., & Lopez-Coronado, M. (2017). Security analysis of a mHealth app in Android: Problems and solutions. *2017 IEEE 19th International Conference on E-Health Networking, Applications and Services (Healthcom)*, 1–6. https://doi.org/10.1109/HealthCom.2017.8210757

Emmanuel Dupuy. (2019). *Java Decompiler*. http://java-decompiler.github.io/

Garg, S., & Baliyan, N. (2021). Android security assessment: A review, taxonomy and research gap study. *Computers & Security*, *100*, 102087. https://doi.org/10.1016/j.cose.2020.102087

Google Play. (2023a). *MyTelkomsel - Buy Package - Apps on Google Play*. https://play.google.com/store/apps/details?id=com.telkomsel.telkomselcm

Google Play. (2023b). *SATUSEHAT Mobile - Apps on Google Play*. https://play.google.com/store/apps/details?id=com.telkom.tracencare

He, Z., Ye, G., Yuan, L., Tang, Z., Wang, X., Ren, J., Wang, W., Yang, J., Fang, D., & Wang, Z. (2019). Exploiting Binary-Level Code Virtualization to Protect Android Applications against App Repackaging. *IEEE Access*, *7*, 115062–115074. https://doi.org/10.1109/ACCESS.2019.2921417

Jung, J.-H., Kim, J. Y., Lee, H.-C., & Yi, J. H. (2013). Repackaging attack on android banking applications and its countermeasures. *Wireless Personal Communications*, *73*(4), 1421–1437. https://doi.org/10.1007/s11277-013-1258-x

Karthick, S., & Binu, S. (2017). Android security issues and solutions. *2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, 686–689. https://doi.org/10.1109/ICIMIA.2017.7975551

Katoch, S., & Garg, V. (2023). Security Analysis on Android Application Through Penetration Testing using Reverse Engineering. *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, 216–222. https://doi.org/10.1109/ICSMDI57622.2023.00048

Martín, I., & Hernández, J. A. (2019). CloneSpot: Fast detection of Android repackages. *Future Generation Computer Systems*, *94*, 740–748. https://doi.org/10.1016/j.future.2018.12.050

May Thu Kyaw, Yan Naung Soe, & Nang Saing Moon Kham. (2019). Security Analysis of Android Application by Using Reverse Engineering. *Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering*. https://doi.org/10.18178/wcse.2019.03.029

Merlo, A., Ruggia, A., Sciolla, L., & Verderame, L. (2021a). ARMAND: Anti-Repackaging through Multi-pattern Anti-tampering based on Native Detection. *Pervasive and Mobile Computing*, *76*, 101443. https://doi.org/10.1016/j.pmcj.2021.101443

Merlo, A., Ruggia, A., Sciolla, L., & Verderame, L. (2021b). You Shall not Repackage! Demystifying Anti-Repackaging on Android. *Computers and Security*, *103*. https://doi.org/10.1016/j.cose.2021.102181

Nadean H. Tanner. (2019). Metasploit. In *Cybersecurity Blue Team Toolkit* (pp. 125–146). John Wiley & Sons, Inc. https://doi.org/10.1002/9781119552963.ch10

Ohello, S. P. (2023). *Development Of Android Malware Evasion Technique For Bypassing Signature-Based Anti-Malware Detection Using Obfuscation And Dynamic Code Loading*. Institut Teknologi Bandung.

Pan, J.-Y., & Ma, S.-H. (2017). Advertisement removal of Android applications by reverse engineering. *2017 International Conference on Computing, Networking and Communications (ICNC)*, 695–700. https://doi.org/10.1109/ICCNC.2017.7876214

Promon. (2022). *App Threat Report: The State of Repackaging*. https://promon.co/app-threat-report-repackaging

Rahalkar, S., & Jaswal, N. (2017). *Metasploit Revealed : Secrets of the Expert Pentester*. Packt Publishing Ltd.

Raj, S., & Walia, N. K. (2020). A Study on Metasploit Framework: A Pen-Testing Tool. *2020 International Conference on Computational Performance Evaluation (ComPE)*, 296–302. https://doi.org/10.1109/ComPE49325.2020.9200028

Rapid7. (n.d.). *Metasploit Framework*. Retrieved October 11, 2023, from https://docs.rapid7.com/metasploit/msf-overview

Rizvi, S., Alden, K., & Campbell, S. (2019). A Hybrid Framework for Detecting Repackaged Applications on the Android Market. *2019 International Conference on Software Security and Assurance (ICSSA)*, 76–82. https://doi.org/10.1109/ICSSA48308.2019.00017

Salem, A., Paulus, F. F., & Pretschner, A. (2018). Repackman: A tool for automatic repackaging of android apps. In J. Klein, G. Meng, S. Malek, & L. Li (Eds.), *A-Mobile 2018 - Proceedings of the 1st International Workshop on Advances in Mobile App Analysis, co-located with ASE 2018* (pp. 25–28). Association for Computing Machinery, Inc. https://doi.org/10.1145/3243218.3243224

Sharma, T., & Rattan, D. (2021). Malicious application detection in android — A systematic literature review. *Computer Science Review*, *40*, 100373. https://doi.org/10.1016/j.cosrev.2021.100373

Sihag, V., Vardhan, M., & Singh, P. (2021). A survey of android application and malware hardening. *Computer Science Review*, *39*, 100365. https://doi.org/10.1016/j.cosrev.2021.100365

Skylot. (2020). *Jadx - Dex to Java decompiler*. https://github.com/skylot/jadx

Song, L., Tang, Z., Li, Z., Gong, X., Chen, X., Fang, D., & Wang, Z. (2018). AppIS: Protect android apps against runtime repackaging attacks. *Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS*, *2017-December*, 25–32. https://doi.org/10.1109/ICPADS.2017.00015

StatCounter Global Stats. (2024). *Mobile Operating System Market Share Indonesia | Statcounter Global Stats*. https://gs.statcounter.com/os-market-share/mobile/indonesia

Statcounter Global Stats. (2024). *Mobile Operating System Market Share Worldwide*. https://gs.statcounter.com/os-market-share/mobile/worldwide

Talsec. (n.d.). *freeRASP in-app protection & security*. Retrieved August 12, 2023, from https://www.talsec.app/freerasp-in-app-protection-security-talsec

Vidas, T., & Christin, N. (2013). Sweetening android lemon markets. *Proceedings of the Third ACM Conference on Data and Application Security and Privacy*, 197–208. https://doi.org/10.1145/2435349.2435378

Wermke, D., Huaman, N., Acar, Y., Reaves, B., Traynor, P., & Fahl, S. (2018). A Large Scale Investigation of Obfuscation Use in Google Play. *Proceedings of the 34th Annual Computer Security Applications Conference*, 222–235. https://doi.org/10.1145/3274694.3274726

Yang, S., Wang, Y., Xu, H., Xu, F., & Chen, M. (2022). An Android Malware Detection and Classification Approach Based on Contrastive Lerning. *Computers & Security*, *123*, 102915. https://doi.org/10.1016/j.cose.2022.102915

Zeng, Q., Luo, L., Qian, Z., Du, X., & Li, Z. (2018). Resilient decentralized Android application repackaging detection using logic bombs. *Proceedings of the 2018 International Symposium on Code Generation and Optimization*, 50–61. https://doi.org/10.1145/3168820

Zheng, X., Pan, L., & Yilmaz, E. (2017). Security analysis of modern mission critical android mobile applications. *Proceedings of the Australasian Computer Science Week Multiconference*, 1–9. https://doi.org/10.1145/3014812.3014814

Zhu, H., Wei, H., Wang, L., Xu, Z., & Sheng, V. S. (2023). An effective end-to-end android malware detection method. *Expert Systems with Applications*, *218*, 119593. https://doi.org/10.1016/j.eswa.2023.119593