IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Challenges in Mobile Application Security

Mrs. Priyanka B. Palve[1], Miss. Shruti Landage[2], Mr. Prem Shirale[3], Mr. Shubham Gayke[4], Miss. Snehal Janephalkar[5]

Prof. Computer Engineering Department, Adsul's Technical Campus, Ahilyanagar, India[1]
Students, Computer Engineering Department, Adsul's Technical Campus, Ahilyanagar, India[2,3,4]
Students, AIDS Engineering Department, Adsul's Technical Campus, Ahilyanagar, India [5]

Abstract: *Over the past year mobile apps have completely changed how we talk to each other, shop, learn, and even handle our money. Phones aren't just for calls anymore. Now, people use their phones as wallets, health trackers, and sometimes even as their main workspace. With everyone depending so much on these apps, security isn't just important it's critical. Mobile apps are very important parts of our daily life. we can't deny any risky nature of any apps. Mobile applications have become an integral part of modern life, enabling users to perform a wide range of activities, from banking and shopping to healthcare and entertainment. However, the widespread adoption of mobile apps has also made them a prime target for cybercriminals, leading to significant security challenges. This article explores the critical importance of mobile application security, highlighting the risks associated with data breaches, financial losses, and reputational damage. It examines key challenges such as device fragmentation, insecure data storage, weak authentication, third-party library risks, and regulatory compliance. The article provides actionable best practices for securing mobile applications, including the adoption of a Secure Development Lifecycle (SDL), data encryption, strong authentication mechanisms, regular security testing, and user education. It also discusses emerging trends in mobile app security, such as AI-driven attacks, 5G vulnerabilities, quantum computing threats, and the role of blockchain and zero-trust architecture. Additionally, the article emphasizes the growing importance of DevSecOps in integrating security into the development pipeline. Through real-world case studies and statistical evidence, this article underscores the need for proactive security measures to protect sensitive user data and maintain trust in mobile applications. By understanding the evolving threat landscape and implementing robust security practices, developers, businesses, and users can build a safer digital ecosystem. The article concludes with a call to action for all stakeholders to prioritize mobile app security in an increasingly connected world.*

Keywords: Mobile Application Security, Data Breach, Authentication, Malware, Reverse Engineering, API Security, Secure Development, Cyber security, phishing attacks, network security

I. INTRODUCTION

Smartphone use has exploded worldwide. With platforms like Android and iOS, we get instant access to apps that handle almost everything—banking, healthcare, shopping, entertainment, you name it. But here's the catch: these apps deal with a ton of personal and financial info, so cyber attackers are always looking for a way in. Mobile devices don't behave like old-school computers. People bounce between networks all the time, hopping onto public Wi-Fi or whatever's available. Plus, a lot of us install third-party apps without even glancing at the permissions. All this just makes it easier for hackers to find a weak spot. This study digs into the biggest security problems facing mobile apps today, and looks at why, even with all our tech progress, these issues just won't go away. Mobile apps make life easier and help us get more done, but they also open the door to all sorts of security risks. Unlike desktop computers, our phones and tablets are always on the move—connecting to public Wi-Fi, talking to all kinds of third-party services. That just means more chances for hackers to sneak [1] These apps deal with a lot of sensitive stuff: your personal info, bank details, where you go, cloud logins, even biometric data like fingerprints. If something goes wrong, you're

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

looking at real problems—identity theft, losing money, corporate spying, or having your privacy invaded. So, if you build, study, or protect mobile apps, you need to understand what you're up against when it comes to security. mobile apps are always talking to backend servers over the internet. If that connection isn't properly encrypted, hackers can grab whatever data gets sent. Weak API security or sloppy server setups make things worse, sometimes leaking reallysensitive info. So, keeping a mobile app secure isn't just about the app itself—it's about locking down the whole network and server side too

Sure, mobile operating systems have gotten better, but applevel security still has some serious gaps. A lot of apps still rely on weak sign-in systems, don't store data safely, leave their APIs exposed, or make it easy for hackers to take them apart. In this paper, I'm digging into the biggest security headaches facing mobile apps and looking at how these problems actually affect people and businesses. The research draws from academic articles, cybersecurity reports, and real-life data breaches. This research paper digs into the big security problems in mobile apps, pulling insights from academic studies, industry reports, and real-world examples. It breaks down why these vulnerabilities happen and looks at how they affect both users and organizations. Turns out, a lot of the trouble comes from sloppy coding, skipping thorough testing, and just not thinking enough about security while building the apps. One big headache is just how scattered mobile operating systems and devices are. With all the different versions of Android and iOS out there, you get a patchwork of security updates and protection features. Lots of folks don't bother keeping their phones up to date, so apps end up exposed to vulnerabilities everyone already knows about. And honestly, developers tend to focus more on making apps look good and run smoothly than locking down security. That means security flaws slip right into the final product. This study really drives home how important it is to tackle mobile app security head-on. Developers, organizations, and users all have to work together—build apps securely, keep them updated, and stay alert to new threats. Strong security frameworks and following the best practices out there actually make a difference; they cut down on risks and keep user data safer from new cyberattacks. In the end, this research gives us a clearer picture of the real challenges out there and points to ways we can build mobile systems that people can actually trust. Another big problem: weak authentication and authorization. A lot of mobile apps use pretty basic logins that don't stand up to brute force attacks, stolen passwords, or even simple phishing scams. When developers don't set up authentication properly, it opens the door for hackers to slip into restricted parts of the app— major security risk right there. And there's more. Malware loves mobile devices. One wrong download and some nasty app start stealing your info, tracking what you do, or taking over your phone. Then you've got reverse engineering and code tampering. Hacker tear into the app's code, searching for weak spots or changing things so they can skirt security rules. Android apps get hit a lot because the platform's so open. On top of that, developers often pull in third-party libraries or APIs without really checking how secure they are. That's like building a house with mystery bricks—you never know what flaws you're bringing in. In the end, the paper makes it clear mobile app security isn't a one-and-done thing. Developers need to stick to secure coding rules, use solid encryption, put proper authentication in place, and keep testing for security issues on a regular basis. If we want to protect user data and keep people trusting digital services, we have to keep raising the bar on mobile app security.

As technology continues to evolve, so must our approach to mobile app security. Emerging trends like blockchain, zero-trust architecture, and DevSecOps offer promising solutions, but they also require ongoing vigilance and adaptation. Ultimately, securing mobile applications is a shared responsibility. Developers must prioritize security in their designs, businesses must invest in robust security measures, and users must adopt safe practices to protect their data. The time to act is now. By working together and staying informed about the latest threats and solutions, we can build a safer digital ecosystem for everyone.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

Fig. 1.Mobile app security challenges data

Discussion:

• The findings of this study provide important insights into the current state of Android application security. The high percentage of insecure data storage and weak cryptographic practices shows that many developers still fail to follow basic security guidelines. Despite the availability of secure APIs such as EncryptedSharedPreferences and Android Keystore, many applications continue to store sensitive data in plain text or external storage.

• The review also reveals that permissions remain one of the most misunderstood security areas. Many apps request dangerous permissions that are unrelated to their core functionality. This not only increases the risk of data leakage but also makes it easier for attackers to exploit apps with broad access rights. Users also tend to approve permissions without proper understanding, making the situation worse.

• The findings also highlight that Inter-Component Communication vulnerabilities remain a major challenge. Misconfigured Activities, Services, and Broadcast Receivers allow attackers to send malicious Intents, hijack app components, or access protected data. Research papers consistently show that developers often forget to protect exported components, which creates an easy entry point for exploitation.

• In terms of malware detection, traditional tools are becoming less effective due to increasing use of code obfuscation, hidden payloads, and dynamic loading techniques. This supports the conclusion that modern defence strategies must include Hybrid Analysis and AI/ML-based detection models. Hybrid frameworks detect both code-level and runtime behaviour issues, offering a more complete security assessment. Machine Learning models like SVM, LSTM, and CNN achieve high accuracy and can detect unknown malware families by learning behavioural patterns.

• The results also emphasize the importance of adopting industry standards such as OWASP MASVS, which provide structured guidelines for secure development. Many vulnerabilities found in the reviewed studies could have been prevented if developers followed even the basic MASVS-L1 requirements.

• Overall, the discussion confirms that Android security is not only a technical challenge but also a process and awareness challenge. Developer training, secure coding education, and automated testing tools must be integrated into the entire Software Development Lifecycle (SDLC) to reduce security risks.

II. LITERATURE SURVEY

• A lot of application just leave sensitive data out in the open, unencrypted. That is a big problem. Then there's sloppy authentication—when apps don't set it up right, hackers can break in with brute-force or credential-stuffing attacks. People studying Android malware have noticed something sneaky: fake apps often copy how real ones look and act, just to trick users. On top of that, some studies show that if developers mess up SSL/TLS settings, attackers can easily eavesdrop on data moving between your phone and the server.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, May 2026 Impact Factor: 8.2

• Another thing—attackers use reverse engineering to pick apart an app's code. That lets them grab API keys or figure out how the app works behind the scenes Permissions are another headache. So many apps ask for way more access than they really need—camera, contacts, microphone, location, you name it. Research shows this isn't just annoying; it's risky. Extra permissions open the door for data misuse. If a sketchy app gets those permissions, it can quietly harvest your info without you noticing.[3]

• Authentication is another weak spot. Relying on simple passwords just doesn't cut it anymore. Studies keep pointing out that weak password rules and missing multi-factor authentication make it way too easy for attackers to brute-force their way in or steal credentials. Some researchers even found that session tokens often don't expire like they should, which means attackers can hijack active sessions. Mobile malware is a big topic too. Analysts have sifted through thousands of malicious apps, and a lot of them look totally normal at first glance. Once installed, though, they quietly collect data or do other shady things. Android seems to get hit more often, probably because of its open ecosystem, but iOS isn't totally safe either.[4]

• Then there's insecure communication. If an app sends data to its server without proper encryption, attackers can intercept it using tricks like Man-in-the-Middle attacks. A lot of research points out that sloppy SSL/TLS setups are a common problem. Reverse engineering is another worry. Attackers can break open app packages and poke around the code. If developers leave secret keys or API credentials in there, it's game over—those secrets are easy to steal if the code isn't obfuscated. Third-party libraries and SDKs also get plenty of attention. Developers use them to add features quickly, but if those components are outdated or vulnerable, they introduce security holes right into the app.

• A lot of research digs into the risks that come with third-party libraries and SDKs. These days, if you're building a mobile app, you're probably leaning on outside libraries to speed things up or add new features. That's fine—until those libraries have security flaws or just aren't kept up to date. Suddenly, your app's got problems you didn't even create. Looking at the big picture, it's clear: sloppy development and weak testing usually open the door to security issues. Researchers don't just suggest better practices—they say you need to bake security into every step, right from the start. Stick to secure coding, check your work with regular security assessments, and you'll dodge a lot of the usual headaches.

III. METHODOLOGY

A. Research Best Practices for Securing Mobile Applications

To mitigate the risks associated with mobile app vulnerabilities, developers and organizations must adopt a proactive and comprehensive approach to security. Below are some of the most effective best practices:

B. Secure Development Lifecycle (SDL) :

Integrating security into every phase of the app development process is crucial.The Secure Development Lifecycle (SDL) ensures that security is considered from the initial planning stages through to deployment and maintenance. This includes threat modeling, secure coding practices, and regular security reviews (Microsoft, 2021). By embedding security into the development process, organizations can reduce vulnerabilities and build more resilient applications.

C. Data Encryption

Encrypting data both at rest and in transit is a fundamental security measure. Strong encryption algorithms such as AES (Advanced Encryption Standard) for data storage and TLS (Transport Layer Security) for data transmission should be used to protect sensitive information from unauthorized access (OWASP, 2023). Encryption ensures that even if data is intercepted or accessed by malicious actors, it remains unreadable and unusable.

D. Strong Authentication and Authorization

Implementing robust authentication mechanisms, such as multi-factor authentication (MFA) and OAuth 2.0, can significantly enhance app security. MFA requires users to provide multiple forms of verification, reducing the risk of unauthorized access. OAuth 2.0 provides a secure framework for authorization, ensuring that only authenticated users can access specific resources (Fett, Küsters, & Schmitz, 2020).

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, May 2026 Impact Factor: 8.2

E. Regular Security Testing

Conducting regular security testing, including penetration testing, vulnerability assessments, and code reviews, helps identify and address potential weaknesses before they can be exploited. Automated tools and manual testing should be combined to ensure comprehensive coverage (Veracode, 2023). Regular testing ensures that security measures remain effective as new threats emerge.

F. Secure APIs

APIs are a critical component of mobile apps, enabling communication between the app and backend services. Ensuring that APIs are authenticated, authorized, and encrypted is essential to prevent unauthorized access and data breaches. API security best practices include using tokens, rate limiting, and validating input data (OWASP, 2023).

G. Use of Trusted Libraries and SDKs

Third-party libraries and SDKs can introduce vulnerabilities if not properly managed. Developers should only use trusted and well-maintained libraries, regularly update dependencies, and remove unused components to minimize risks (Synopsys, 2022). Tools like dependency checkers can help identify and address vulnerabilities in third-party code.

H. User Education

Educating users on secure practices is an often-overlooked aspect of mobile app security. Users should be encouraged to download apps only from official stores, avoid sideloading, and recognize phishing attempts. Providing clear guidelines and warnings can help users make informed decisions and reduce the risk of compromise (Kaspersky, 2022).

I. Compliance with Regulations Staying compliant with data protection regulations such as GDPR, CCPA, and HIPAA is essential for avoiding legal penalties and maintaining user trust. Organizations should implement necessary controls, such as data anonymization and access logging, to meet regulatory requirements (GDPR Enforcement Tracker, 2021).

J. Monitoring and Incident Response Continuous monitoring of mobile apps for suspicious activity is critical for detecting and responding to threats in real-time. Organizations should also have a well-defined incident response plan in place to quickly address security breaches and minimize damage (IBM Security, 2022).

IV. TOOLS & TECHNOLOGY FOR MOBILE APP SECURITY

A variety of tools and technologies are available to help developers and organizations secure their mobile applications. These tools address different aspects of security, from code analysis to threat detection. Tools for Static and Dynamic Analysis • MobSF (Mobile Security Framework): An open-source toolor automated security testing of mobile apps, supporting both static and dynamic analysis (MobSF, 2023). • OWASP ZAP (Zed Attack Proxy): A dynamic application security testing (DAST) tool that helps identify vulnerabilities in web services and APIs used by mobile apps (OWASP, 2023).

A. Role of AI and Machine Learning in Threat Detection

Artificial intelligence (AI) and machine learning (ML) are increasingly being used to enhance mobile app security. These technologies can analyze vast amounts of data to detect anomalies, predict potential threats, and automate responses. For example, AI-powered systems can identify unusual user behavior or detect malware in real-time (McAfee, 2023). Importance of Secure Backend Systems and Cloud Services Mobile apps often rely on backend systems and cloud services to store and process data. Ensuring the security of these components is just as important as securing the app itself. Best practices include using secure APIs, encrypting data in transit, and implementing robust access controls (Amazon Web Services, 2023). S

B. Future Trends in Mobile Application Security

As technology evolves, so do the threats and solutions in mobile application security. Staying ahead of emerging trends is crucial for developers, businesses, and users to ensure robust protection against future risks. This section explores the most significant trends shaping the future of mobile app security, including emerging threats, advancements in security technologies, and the growing role of DevSecOps.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, May 2026 Impact Factor: 8.2

Emerging Threats AI-Driven Attacks: Cybercriminals are increasingly leveraging artificial intelligence (AI) to launch sophisticated attacks. AI-driven malware can adapt to security measures, evade detection, and exploit vulnerabilities more efficiently than traditional methods. For example, AI-powered phishing campaigns can generate highly personalized messages that are difficult to distinguish from legitimate communications (McAfee, 2023). Implications: Mobile apps must integrate AI-driven defense mechanisms, such as anomaly detection and behavioral analysis, to counter these advanced threats.

5G Vulnerabilities: The rollout of 5G networks introduces new security challenges, such as increased attack surfaces and vulnerabilities in network slicing and edge computing. For instance, the distributed nature of 5G networks can make it harder to detect and mitigate attacks on mobile apps (Ericsson, 2023).

Implications: Developers must adopt secure coding practices and work closely with network providers to address 5Gspecific vulnerabilities. Quantum Computing Threats: While still in its infancy, quantum computing poses a future threat to encryption algorithms. Mobile apps relying on current encryption standards, such as RSA and ECC, may need to adopt quantum-resistant algorithms like latticebased cryptography to stay secure (NIST, 2023).

Implications: Organizations should start planning for the post-quantum era by exploring quantum-resistant encryption methods and updating their security protocols.

C. Advancements in Security Technologies

Blockchain for Data Integrity: Blockchain technology is being explored as a way to enhance data integrity and secure transactions in mobile apps. Its decentralized nature makes it resistant to tampering and fraud. For example, blockchain can be used to create immutable logs of user transactions, ensuring transparency and accountability (IBM, 2023). Applications: Blockchain is particularly useful in industries like finance, healthcare, and supply chain management, where data integrity is critical. Zero-Trust Architecture: The zero-trust model, which assumes no user or device is inherently trustworthy, is gaining traction. Implementing zero-trust principles in mobile apps ensures continuous verification and minimizes the risk of unauthorized access. For instance, zero-trust can enforce strict access controls based on user behavior and device health (Forrester, 2023). Applications: Zero-trust is ideal for organizations with remote workforces or those handling highly sensitive data. Biometric Authentication: Advances in biometric technologies, such as facial recognition, fingerprint scanning, and voice authentication, are improving authentication mechanisms. These methods are more secure and userfriendly than traditional passwords. For example, Apple's Face ID and Touch ID have set new standards for biometric security in mobile apps (Gartner, 2023). Applications: Biometric authentication is particularly useful in banking, healthcare, and government apps, where security and convenience are paramount.

D. The Role of DevSecOps

DevSecOps, which integrates security into the DevOps pipeline, is becoming essential for mobile app development. By embedding security practices into continuous integration and continuous deployment (CI/CD) pipelines, organizations can identify and address vulnerabilities earlier in the development process. Key components of DevSecOps include: Automated Security Testing: Tools like static application security testing (SAST) and dynamic application security testing (DAST) can automatically scan code for vulnerabilities during development. Code Analysis: Regular code reviews and automated analysis tools help ensure that secure coding practices are followed. Compliance Checks: Automated compliance checks ensure that apps meet regulatory requirements, such as GDPR and HIPAA, throughout the development lifecycle (GitLab, 2023). Benefits: DevSecOps reduces the time and cost of fixing vulnerabilities, improves collaboration between development and security teams, and ensures that security is a priority from the start.

V. RESULT

The results illustrate the major security vulnerabilities found in mobile applications. The analysis shows that insecure data storage (32%) is the most common security issue, where sensitive user data is stored without proper encryption or

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

protection. Weak authentication (24%) is another significant vulnerability that allows unauthorized users to access mobile applications due to poor login or verification mechanisms. Additionally, insecure communication (18%) occurs when applications transmit data over unprotected networks, making them vulnerable to interception attacks. Reverse engineering (14%) is also a major risk, where attackers analyse application code to discover weaknesses. Furthermore, third-party risks (12–14%) arise when external libraries or APIs introduce security flaws.

Fig. 2.Analysis of mobile app vulnerabilities

Fig. 3. This pic illustrates various security issues happens on apps

VI. CONCLUSION

Mobile apps are everywhere now. People use them for pretty much everything — banking, shopping, healthcare, chatting with friends, you name it. We depend on these apps a lot, both at work and in our personal lives. But as we lean more on mobile apps, the security risks keep piling up. This research makes it clear: mobile app security isn't just some technical box to check. It's vital for protecting people's data and keeping their trust. The study spotted a bunch of big security problems. Stuff like insecure data storage, weak logins, sketchy communication channels, reverse engineering, API loopholes, and even risks hiding in third-party libraries. These gaps aren't just harmless bugs — they open the door to things like financial fraud, identity theft, leaked data, and privacy nightmares. And honestly, a lot of these issues don't even come from super-sophisticated hackers. More often, it's sloppy coding, skipping security tests,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, May 2026
Impact Factor: 8.2

or just not paying enough attention during development. [8] One of the biggest problems? Teams still treat security like it's an add-on, not a must-have. Developers usually pour their energy into making the app run smoothly or look good, but security testing ends up on the back burner. That means vulnerabilities get missed until somebody actually launches an attack. The study also points out how much backend systems and APIs matter for security. Even if the app itself looks locked down, a weak server setup can leak sensitive info. Because these apps handle personal and financial information, security is very important. This research shows that many mobile apps still have common security problems. These include weak passwords, poor data storage, insecure communication, and unsafe use of third-party libraries. Most of these issues happen because of poor coding practices and lack of proper security testing.The study also points out that mobile security isn't just on one person or team. Developers have to write secure code and stick to security guidelines. Organizations need to run regular security tests and actually invest in good protection systems. And users? They've got their part to play too—only downloading apps from trusted sources, setting strong passwords, and keeping their devices up to date. If any one of these groups drops the ball, the whole system is at risk. Finally, mobile app security isn't something you set and forget. Threats keep changing, and attackers always come up with new tricks. So, mobile security strategies need to keep up. That means constant monitoring, regular updates, checking for vulnerabilities, and running security audits. It's the only way to keep long-term risks under control.[10]

Mobile application security is a critical concern in today's digital landscape, where apps handle sensitive data, facilitate financial transactions, and play a central role in daily life. The challenges are numerous, ranging from device fragmentation and insecure coding practices to emerging threats like AIdriven attacks and 5G vulnerabilities. However, by adopting best practices such as secure development lifecycles, data encryption, strong authentication, and regular security testing, developers and organizations can significantly mitigate these risks.

ACKNOWLEDGMENT

It gives us great pleasure in presenting the paper on "Challenges in Mobile Application Security". We would like to take this opportunity to thank our guide, Prof.RriyankaPalve, Professor, Computer Department, Adsul's technical Campus, Ahlyanagar, for giving us all the help and guidance we needed. We are grateful to her for hers kind support, and valuable suggestions were very helpful.

REFERENCES

[1]. D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, "Drebin: Effective and explainable detection of Android malware in your pocket," in Proc. NDSS, 2014

[2] M. Conti, V. T. N. Nguyen, and B. Crispo, "Mobile application security: A comprehensive review," IEEE Communications Surveys & Tutorials, vol. 17, no. 2, pp. 102–124, 2015.

[3] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner, "Android permissions demystified," in Proc. ACM CCS, 2012, pp. 627–638.

[4] W. Enck, M. Ongtang, and P. McDaniel, "On lightweight mobile phone application certification," in Proc. ACM CCS, 2009, pp. 235–245.

[5] OWASP Foundation, "OWASP Mobile Top 10 – 2023," [Online]. Available: https://owasp.org.

[6] OWASP Foundation, "Mobile Security Testing Guide (MSTG)," 2023

[7] Check Point Research, "Mobile Security Report 2023," Check Point Software Technologies, 2023.

[8] IBM Security, "Cost of a Data Breach Report 2023," IBM Corporation, 2023.

[9] S. Kumar and K. Dutta, "Mobile cloud computing: Security issues and challenges," Journal of Information Security and Applications, vol. 42, pp. 123–135, 2018.

[10] V. Rastogi, Y. Chen, and X. Jiang, "DroidChameleon: Evaluating Android anti-malware against transformation attacks," in Proc. ACM Asia CCS, 2013, pp. 329–334.

[11] A. Shabtai, Y. Fledel, and Y. Elovici, "Securing Androidpowered mobile devices using SELinux," IEEE Security & Privacy, vol. 8, no. 3, pp. 36–44, 2010