Android Source Code Vulnerability Detection: A Systematic Literature Review

JANAKA SENANAYAKE, Robert Gordon University, UK and University of Kelaniya, Sri Lanka
HARSHA KALUTARAGE, Robert Gordon University, UK
MHD OMAR AL-KADRI, Birmingham City University, UK
ANDREI PETROVSKI, Robert Gordon University, UK
LUCA PIRAS, Middlesex University, UK

The use of mobile devices is rising daily in this technological era. A continuous and increasing number of mobile applications are constantly offered on mobile marketplaces to fulfil the needs of smartphone users. Many Android applications do not address the security aspects appropriately. This is often due to a lack of automated mechanisms to identify, test, and fix source code vulnerabilities at the early stages of design and development. Therefore, the need to fix such issues at the initial stages rather than providing updates and patches to the published applications is widely recognized. Researchers have proposed several methods to improve the security of applications by detecting source code vulnerabilities and malicious codes. This Systematic Literature Review (SLR) focuses on Android application analysis and source code vulnerability detection methods and tools by critically evaluating 118 carefully selected technical studies published between 2016 and 2022. It highlights the advantages, disadvantages, applicability of the proposed techniques, and potential improvements of those studies. Both Machine Learning (ML)-based methods and conventional methods related to vulnerability detection are discussed while focusing more on ML-based methods, since many recent studies conducted experiments with ML. Therefore, this article aims to enable researchers to acquire in-depth knowledge in secure mobile application development while minimizing the vulnerabilities by applying ML methods. Furthermore, researchers can use the discussions and findings of this SLR to identify potential future research and development directions.

CCS Concepts: • **Security and privacy → Software security engineering**; **Software reverse engineering**; Domain-specific security and privacy architectures; Malware and its mitigation; **Vulnerability scanners**; Mobile platform security; • **Computing methodologies → Machine learning;**

Additional Key Words and Phrases: Source code vulnerability, vulnerability detection, software security, Android security, machine learning

Authors' addresses: J. Senanayake, Robert Gordon University, Garthdee Road, Aberdeen, UK, AB10 7QB, and University of Kelaniya, Dalugama, Kelaniya, Western Province, Sri Lanka, 11600; email: j.senanayake@rgu.ac.uk, janakas@kln.ac.lk; H. Kalutarage and A. Petrovski, Robert Gordon University, Garthdee Road, Aberdeen, UK, AB10 7QB; emails: {h.kalutarage, a.petrovski}@rgu.ac.uk; Mhd O. Al-Kadri, Birmingham City University, Millennium Point, Curzon Street, Birmingham, UK, B4 7XG; email: omar.alkadri@bcu.ac.uk; L. Piras, Middlesex University, The Burroughs, London, UK, NW4 4BT; email: l.piras@mdx.ac.uk.