Centro Sur Vol. 9 No. 4- October - December - Revista Centro Sur - eISSN: 2600-5743

http://centrosureditorial.com/index.php/revista

The Trojan horse in your pocket: How an innocent app can empty your bank account

El caballo de Troya en tu bolsillo: Cómo una app inocente puede vaciar tu cuenta bancaria

José Luis Hidalgo Torres
Process Management Engineer
Edwards Deming Corporate University Technological Institute
jlhidalgot@hotmail.com
https://orcid.org/0000-0002-4671-3116

ABSTRACT

In the digital age, mobile security is an extremely critical challenge given the growing proliferation of financial malware and sophisticated social engineering. This article explores how common mobile applications can contain sophisticated mechanisms for illegitimate financial extraction that exploit systematic abuses of critical permissions, overlay attacks, and stealthy credential capture. In summary, a combined static and dynamic analysis of malicious Android APKs is performed with the help of specialized tools such as Sandboxing, Wireshark, and permission analysis on Android devices. Common patterns of abuse of accessibility services and overlay techniques used to implement highly realistic banking interfaces are presented, with key examples of malware such as BianLian and SharkBot. With this approach to the threats presented and considering the options available to counter them. Strategies must be directed at users in executable ways, especially critical permission auditing and

https://doi.org/10.37955/cs.v9i3.396
Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

the identification of signs of social engineering. In conclusion, users must seek to protect themselves and application developers.

RESUMEN

En la era digital, la seguridad móvil es un desafío sumamente crítico dada la creciente proliferación de malware financiero y la sofisticada ingeniería social. Este artículo explora cómo las aplicaciones móviles usuales pueden contener mecanismos sofisticados para la extracción financiera ilegítima y que explotan abusos sistemáticos de permisos críticos, los ataques de superposición y la captura de credenciales de manera furtiva. En resumen, se realiza un análisis combinado, estático y dinámico, de APK maliciosas de Android, con la ayuda de herramientas especializadas, tal como Sandboxing, Wireshark, y el análisis de permisos en dispositivos Android. Se presentan los patrones comunes de abuso de los servicios de accesibilidad y las técnicas de superposición utilizadas para implementar interfaces bancarias con gran realismo, con ejemplos principales de malware, tal como BianLian y SharkBot. Con este enfoque respecto a las amenazas presentadas y considerando las opciones disponibles para contrarrestarlas. Las estrategias hay que direccionarlas a los usuarios en formas ejecutables, especialmente la auditoría crítica de permisos y la identificación de signos de ingeniería social. Para concluir, los usuarios deben buscar protegerse a sí mismos y los desarrolladores de la aplicación.

Keywords

Mobile security, banking malware, social engineering, application permissions, malware prevention.

Seguridad móvil, malware bancario, ingeniería social, permisos de aplicaciones, prevención de malware.

INTRODUCTION

In the age of hyperconnectivity, the smartphone has gone beyond its original function to become a crucial extension of everyday life. From the ability to communicate effectively to conducting financial transactions, accessing digital identities, and services provided, the smartphone is nothing more than a link between our digital and physical inaction. In line with this reality, it is reported that more than

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

7 out of 10 online banking and commercial transactions are conducted using mobile devices (Feedzai, 2025).

However, this growing dependence also creates vulnerabilities, mainly due to the uncontrolled proliferation of mobile applications. Many of these applications, even those that appear harmless and legitimate, such as calculators, games, and utilities, can hide malicious mechanisms that operate silently beneath their innocent appearance. This is where the metaphor of the "Trojan horse" comes in; they are a camouflaged and seemingly harmless threat, but in reality they can and are capable of emptying the user's bank account in a matter of seconds without them noticing.

The problem is exacerbated by the frequent neglect of good security practices, both by developers and users. From the engineering of applications that request excessive permissions without clear justification to users who accept without question, an environment conducive to the proliferation of mobile financial malware is created. Malware such as BianLian and SharkBot in particular have proven to be surprisingly efficient, combining advanced techniques to modify program code (obfuscation), abuse permissions, overlay interfaces, and use social engineering-based psychological manipulation to achieve their goals (INCIBE, 2025).

A phenomenon of this nature requires analysis that scrutinizes not only the technical nature of the malware in terms of how it works, what APK permissions it obtains, what type and complexity of obfuscation is applied, and the user profile. Social engineering plays a crucial role in making the attack work, and even though most users have basic training, they end up falling into the trap (Malwarebytes, 2025). For example, a malicious application that requires authorization for accessibility services may justify a reason for users with disabilities, but in reality it is used to load the legitimate bank interface and request credentials. Therefore, an interdisciplinary approach is taken and a comprehensive and up-to-date assessment of the threat of malicious financial applications on Android is proposed. Through technical analysis and analysis of the respective social impact, we seek to answer essential questions: How do these crimes work technically? What are the most critical vulnerabilities? What role do deception techniques play? And how can attacks be successfully prevented with publicly available tools and basic preparation? The main objectives are:

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

- To describe in detail the technical characteristics and evolution vectors of the most recent and widespread malicious financial APKs.

- To analyze the use and impact of social engineering in the promotion and effectiveness of these crimes.

To offer practical, zero-cost recommendations for users and developers that, when implemented, minimize risk and foster a culture of responsible defense against the malicious use of mobile devices. The aim is not only to raise awareness, but also to contribute concrete solutions to the challenge of preventing and mitigating a threat that, although less visible, has extreme financial and personal consequences for millions of users around the world.

Methodology

Taking a comprehensive approach to how malicious mobile applications that act as digital Trojan horses operate, a mixed methodological approach was adopted, focusing on technical malware analysis and a specialized documented review of the state of mobile financial threats. This mixed methodological approach allowed for the simultaneous analysis of the technical characteristics of malicious APKs and the contextualization of their impact based on the community's experience with the simultaneous use of certain computer security conventions and social engineering.

Representative samples of mobile banking malware such as BianLian and SharkBot, which are currently prevalent and high-profile in the computer security community, were selected for analysis. This malware has received considerable publicity in the community for extended periods due to its high technical sophistication, use of exceptional evasion measures, and high number of infections spread, having had an impact in multiple regions, including Europe, South America, and Asia (INCIBE, 2025), (SECURESOFT, 2022).

The samples used were obtained and validated through public repositories and security analysis databases certified by recognized entities in mobile threat research, ensuring their relevance and representativeness in the current threat landscape (SECURELIST, 2025), (Kaspersky, 2025). For example, platforms such as ThreatFabric and Kaspersky reports continuously document the activity and evolution of this malware, providing access to samples and

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

behavior patterns that support the article (Infobae, 2025), (SECURELIST, 2025).

This ensures that the technical analysis is based on real, current, and representative evidence, allowing valid conclusions to be drawn that are applicable to the prevention of contemporary mobile financial attacks.

Static analysis is an essential methodology in mobile banking malware research, allowing the structure, code, and internal components of an application to be studied without the need to execute it, minimizing risks and facilitating the accurate identification of malicious elements (Rojo, 2020), (Rivera, 2014).

This analysis is based on reverse engineering techniques and the use of specialized tools that decompile the APK to extract valuable information about permissions, code, and integrated resources (UPS, 2025).

The main objective of static analysis is to identify indicators such as requested permissions, declared manifests, hidden components, suspicious text strings, and obfuscated code that reveal the malicious nature of the application (Ismael & Thanoon, 2022), (Rivera, 2014).

APK decompilation:

Tools such as APKTool and JADX are used to extract and examine the Android Manifest.xml file to identify permissions such as READ_SMS, SEND_SMS, SYSTEM_ALERT_WINDOW, and others that could be used to intercept communications or modify interfaces (Rojo, 2020).

Services and receivers operating in the background for malicious purposes are also identified (UPS, 2025).

String and Resource Analysis:

Extraction of text strings to detect URLs, IP addresses, commands, or data that point to connections with control and tracking servers (Appdome, Inc, 2025).

Java and Dalvik Code Review:

Manual and automatic inspection of calls to sensitive APIs, use of obfuscation techniques, dynamic encryption, and content loading in

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Web Views used for overlay attacks and credential theft (Gaviria, 2016), (Rivera, 2014).

Certificate validation:

Review of digital signatures to detect modifications or inclusion of unauthorized third-party components.

Native library analysis:

Where applicable, native code analysis is performed on files. Using disassemblers to detect low-level malicious instructions (Rojo, 2020).

Static analysis relies on the following tools, which are widely recognized in the malware analysis community:

- APKTool for disassembling resources and manifests (Rivera, 2014).
- JADX, JEB, CFR to decompile Dalvik byte code to readable Java code (Rojo, 2020).
- Appdome for automation in static and dynamic analysis (Appdome, Inc, 2025).
- Androguard for in-depth analysis with Python scripting (Gaviria, 2016).
- Virus Total for scanning with multiple antivirus engines (Rivera, 2014).

Static analysis allows for the detection of known malware, performs permission audits, and understands the overall structure prior to execution; however, it is limited by advanced encryption and obfuscation techniques that make it difficult to read the code directly (Ismael & Thanoon, 2022), (Rivera, 2014).

For this reason, it is complemented by dynamic analysis to validate behaviors during execution (Rojo, 2020).

Static analysis facilitated the identification of characteristic patterns in financial malware such as BianLian and SharkBot, confirming their abuse of critical permissions and the incorporation of sophisticated techniques to camouflage their malicious activity.

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Dynamic analysis is an essential part of the mobile banking malware analysis process, allowing the behavior of malicious software to be observed and controlled during its execution in a controlled environment, revealing actions and effects that are not detectable through static analysis (UPS, 2025), (Granado-Masid, 2012).

The purpose of this technique is to understand the actual functionality of the malware by monitoring its processes in real time, file system modifications, system calls, network communications, and other indicators of malicious activity (Granado-Masid, 2012), (University of Seville, 2020).

Creation of controlled environments (sandboxes):

A secure laboratory is set up, with isolated virtual machines that are not connected to the production network, to run the malware and prevent contagion or collateral damage. In this environment, real-world conditions are simulated to capture the natural behavior of the malware (UPS, 2025).

Monitored execution:

Monitoring tools are used to record activities such as file creation or deletion, registry changes, processes started, network calls, and resource usage. This detailed monitoring allows the malware's lifecycle and functionality to be mapped in real time (Granado-Masid, 2012).

Network traffic analysis:

Traffic generated by malware is captured to identify communications with command and control servers, transfer of stolen data, and possible mechanisms for updating or downloading additional payloads (University of Seville, 2020).

Instrumentation and debugging:

Tools such as Frida allow functions to be intercepted and execution to be modified in order to discover hidden code or trigger specific behaviors on demand. Other debuggers and code analyzers, such as IDA Pro and Wireshark, complement the observation (Granado-Masid, 2012).

Tools used

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Sandboxes (e.g., Cuckoo Sandbox, Appdome): to run malware in secure environments and manage results (Appdome, Inc., 2025).

Wireshark: for capturing and detailed analysis of network traffic (UPS, 2025).

Frida: for dynamic code instrumentation and runtime manipulation (Granado-Masid, 2012).

IDA Pro, OllyDbg: debuggers for in-depth analysis and code reversal (University of Seville, 2020).

Snort and other IDS: for pattern-based intrusion detection during simulation (University of Seville, 2020).

Dynamic analysis provides a comprehensive view of the actual behavior of malware, detecting malicious actions that remain hidden in static analysis. However, it is costly in terms of time and resources, and some malware includes anti-sandbox and anti-debugging techniques to avoid detection during this phase (Granado-Masid, 2012), (ESET AWARDS, 2017).

Samples of BianLian and SharkBot were run in a laboratory equipped with sandboxing and network monitoring techniques. Characteristic behaviors were observed, such as the superimposition of fake banking interfaces, the interception of SMS messages, and the automatic sending of credentials to remote servers, confirming and complementing the results of the static analysis (Rojo, 2020).

Social engineering is a fundamental technique that cybercriminals use to successfully compromise mobile devices with banking malware. This strategy is based on psychologically manipulating the user, causing them to compromise their security by downloading malicious applications or granting critical permissions. What is disturbing about this technique is that it can be carried out without directly breaching the system's technical defenses (Malwarebytes, 2025), (Kaspersky, 2025).

This type of attack plays on human requirements, such as trust, urgency, and fear, to manipulate the user into revealing sensitive information or performing actions that will benefit the attacker.

Some of the most common methods used are:

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Phishing and smishing: Through emails or text messages that appear to come from a trusted entity, such as a bank, the user is invited to click on dangerous links or download applications that falsely promise solutions or urgent alerts (Godoy, 2024), (Infobae, 2025).

Vishing: In this case, the attacker makes phone calls posing as a bank employee or technician, with the aim of obtaining confidential data directly from the victim. (LISA Institute, 2025).

Identity theft and fraudulent websites: Here, attackers create websites or applications that perfectly mimic the original versions in order to capture credentials and private keys without the user noticing (IBM, 2022), (Malwarebytes, 2025).

Exploitation of accessibility services: Attackers request accessibility permissions under seemingly legitimate pretexts, but then use them for malicious activities, such as overlaying fake login screens that steal banking details (ABOUT FRAUD, 2025).

You have to be alert and cautious to avoid falling into these traps.

We have seen how NGate malware has caught the attention of experts such as ESET, as it combines social engineering techniques with advanced technical capabilities. In this case, victims receive fake SMS messages that appear to be official communications, leading them, without realizing it, to install malicious applications from external links. This allows attackers to access their banking details and contact list (ESET, 2024).

Another worrying case is that of Crocodilus malware, which is mostly spread through misleading social media campaigns and false advertising. Scammers trick users into downloading fraudulent applications by presenting them with attractive promises. Once the victim installs the application, the attackers manipulate permissions to gain remote control of the device, allowing them to empty people's bank accounts (Infobae, 2025).

Social engineering is more complicated than technical vulnerabilities, playing on human psychology, an issue that cannot be managed solely with technological systems. With this logic, user education and awareness are essential to reducing risks (Malwarebytes, 2025), (LISA Institute, 2025).

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Attackers are very skilled at creating credible and urgent situations, which leads us to need a cultural change that encourages a critical and suspicious attitude towards unexpected requests. In addition, it is essential that we are rigorous in managing permissions and access in our mobile applications.

Results

An analysis of BianLian and SharkBot reveals a series of technical features and behaviors that explain how these malicious applications are so effective at draining bank accounts and evading traditional security systems.

Technical behavior and permission abuse

In BianLian and SharkBot, we can see that, from the moment they are installed, they request excessive permissions related to Android accessibility services. These permissions allow significant control over the device interface. Thanks to this abuse, the malware can create overlay windows that mimic the authentic screens of banking applications. This causes users, in a moment of carelessness, to enter their credentials and authorizations without realizing that they are being deceived (INCIBE, 2025).

In addition, this malware has been found to have particularly invasive functions. It can record keystrokes, capturing sensitive information entered by the user. It also has the ability to intercept and hide SMS messages, including authentication codes, facilitating unhindered access to accounts. Another tactic they use is to execute USSD commands, allowing them to check balances or perform operations without the user's consent.

To add an extra level of control, they can lock the device's screen while carrying out the attack, making it difficult for the victim to detect or interrupt. Finally, they have the ability to download and execute additional payloads, allowing them to extend the damage caused or maintain their presence on the device remotely.

Advanced evasion techniques

Both types of malware have developed very ingenious techniques to evade detection and analysis. For example, they incorporate systems

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

that allow them to evade antivirus software and have anti-analysis mechanisms that complicate the task of inspecting their behavior in sandbox environments. This makes it difficult for security solutions to recognize their signatures (La Vanguardia, 2022). In addition, they employ encryption in their code and use encrypted communication with their command and control servers, making them even more difficult to track and block by conventional security tools.

Modus operandi and infection vectors

Applications that harbor malware often disguise themselves as common and useful tools, such as currency calculators, file cleaners, or music players. In this way, they try to gain the user's trust in order to deceive them, something we have seen in cases such as SharkBot and BianLian. Furthermore, it is worrying that these malicious programs use phishing campaigns, fake advertisements, and manipulated positive reviews, all with the aim of infecting as many devices as possible (INCIBE, 2025), (La Vanguardia, 2022).

Once these malicious applications are installed and obtain the necessary permissions, they can take control of the device to perform harmful activities. On the one hand, they carry out attacks known as ATS (Automatic Transfer Systems), which allow them to make unauthorized transfers by exploiting vulnerabilities in multi-factor authentication systems. In addition, they can access and manipulate the user's bank accounts by connecting to money mule services managed by the attackers. All of this circumvents standard security procedures, such as adding new devices to a whitelist, further complicating fraud detection (INCIBE, 2025), (La Vanguardia, 2022).

Impact on users

Users and security experts have shared several worrying experiences regarding these attacks. Among the most common effects, many people have seen their accounts quickly emptied, resulting in considerable financial losses. In addition, during an attack, some have completely lost control of their devices, which can be distressing. It is also common for phones to become much slower than usual, and for users to suddenly notice an unexplained increase in data and battery consumption. Finally, users have reported receiving fraudulent notifications that block their legitimate interactions, further complicating the situation.

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

The findings indicate that mobile financial malware is becoming increasingly sophisticated, posing a real and growing threat to user security and the integrity of the digital financial system.

Of particular concern is the misuse of permissions to access accessibility services. This suggests that these malicious programs are exploiting existing vulnerabilities in the Android structure to maintain their presence and take control of users' devices.

This phenomenon reflects what has been warned about in global reports from 2024 and 2025, which document an alarming 102% increase in the number of users affected by mobile financial threats (Kaspersky, 2025).

While traditional attacks targeting PCs appear to be on the decline, the growing presence and use of mobile devices has led cybercriminals to shift their focus. These criminals are using increasingly sophisticated techniques, such as code obfuscation and encrypted communication, as well as detection schemes made more difficult by the use of artificial intelligence. This allows them to maintain greater resilience and effectiveness, as seen in malware such as BianLian and SharkBot (SECURELIST, 2025).

From a social engineering perspective, psychological manipulation plays a key role in the success of attacks. Cybercriminals are experts at disguising malicious applications as everyday tools. These applications are often accompanied by urgent messages, false security warnings, and misuse of permissions, which can deceive even those users who are aware of digital risks (Malwarebytes, 2025). This human aspect reveals that technological barriers alone are not enough to protect us. For this reason, it is vital to include educational and awareness strategies that help users develop critical thinking and a more cautious approach to digital security.

It is important to recognize that this research has some limitations. One of the main ones is the changing and constantly evolving nature of mobile malware, which can quickly alter its attack methods and strategies to evade detection. This means that our analysis will always be in the process of being updated. Furthermore, regional variations and attacks targeting specific groups were not explored. Therefore, it would be valuable for future articles to consider broader samples and longitudinal approaches, which would help us better understand how these threats spread and mutate over time.

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

In practice, this article suggests that the best way to address the challenges of mobile malware is to adopt an approach that includes multiple dimensions. This involves:

Developing secure applications, which means granting only the permissions that are strictly necessary and conducting rigorous audits of applications that are public.

It is essential to implement new technologies that use machine learning to detect unusual behavior and communication that could be hiding.

In addition, it is important to continuously educate users about social engineering. It is essential that they understand the risks they face, know how to identify warning signs, and responsibly manage permissions on their mobile devices, contributing to a better defense against malware.

Mobile platforms and app stores need to strengthen their regulations to carry out more rigorous review and certification of apps to prevent malicious apps from entering the market.

In summary, we have a complicated landscape where technical failures, increasingly sophisticated malware, and human vulnerabilities are intertwined. To address this situation, it is essential to foster collaboration between different disciplines, drive technological innovation, and maintain continuous education for society. This will enable us to effectively address these stealthy technological challenges.

Conclusions

This article clearly shows how malicious mobile applications have become a veritable digital "Trojan horse," putting the financial security of millions of users at risk. By combining advanced social engineering techniques with the misuse of crucial permissions in the Android operating system, malicious programs such as BianLian and SharkBot are able to circumvent traditional security protections. As a result, they steal banking credentials directly, forge legitimate interfaces, and even carry out financial transactions without users' consent. This poses a serious challenge to the protection of our personal and financial information in the digital world.

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

The article makes us reflect on the importance of security on mobile devices from a broad perspective, focusing not only on technology but also on people's behaviors and decisions. It is essential that developers and users be responsible when granting permissions to applications. This, along with the use of effective tools to detect and monitor potential threats, could help significantly reduce the associated risks. In addition, promoting ongoing education about social engineering is essential to empower users to recognize and avoid these threats, thus protecting their personal and financial information in an increasingly complex digital environment.

In conclusion, it is essential that more rigorous regulations and oversight be implemented in mobile app markets. It is also crucial to encourage collaboration between the public and private sectors to lessen the negative impact of these threats on today's digital economy. This will allow us to protect our data and create a more secure environment for managing finances from mobile devices.

References

About Fraud. (2025). *ThreatFabric* - Fraud Solution Profile: https://www.about-fraud.com/providers/threatfabric/

Appdome, Inc. (2025). *Mobile Application Security*. Mobile Application Security Without the Headaches: https://www.appdome.com/mobile-app-security/?utm_medium=ppc&utm_term=mobile%20app%20security&utm_campaign=LATAM_Ecuador_Mobile_Security_Fraud_Security&utm_source=adwords&hsa_kw=mobile%20app%20security&hsa_cam=20273166694&hsa_ver=3&hsa_acc=5868061849&hsa_a

ESET. (2024). *Discover malware for Android that relays NFC traffic to steal money from victims at ATMs*: https://www.eset.com/py/acerca-de-eset/sala-de-prensa/comunicados-de-prensa/articulos-de-prensa/eset-descubre-un-malware-para-android-que-retransmite-trafico-nfc-para-robar-dinero-de-las-victimas-en-cajeros-automaticos/

Feedzai. (2025). *Malware analysis*: SharkBot: https://www.feedzai.com/es/material/analisis-de-malware-sharkbot/

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

Gaviria, P. (2016). *Re-Unir*. Application of Malware Methodology for the Analysis of the Advanced Persistent Threat (APT) "Poison Ivy": https://reunir.unir.net/bitstream/handle/123456789/4738/GAVIRIA%20%2C%20PABLO%20ANDRES.pdf?sequence=1&isAllowed=y

Godoy, L. (2024). *El Comercio*. Cyberattacks Using Social Engineering: https://www.elcomercio.com/opinion/ciberataques-ingenieria-social-lorena-naranjo-columnista/

Granado-Masid, A. (2012). *Reunir - Digital repository*. Application of a malware analysis methodology to a sample of ransomware: https://reunir.unir.net/handle/123456789/12261

IBM. (2022). *What is social engineering*?: https://www.ibm.com/es-es/topics/social-engineering

INCIBE. (2025). *Incibe_National Cybersecurity Institute*. Sharkbot: https://www.incibe.es/servicio-antibotnet/info/Sharkbot

Infobae. (2025). *New security threat on Android*: this is the malware that targets bank accounts: https://www.infobae.com/tecno/2025/06/30/nueva-amenaza-de-seguridad-en-android-asi-es-el-malware-que-va-tras-las-cuentas-bancarias/

Ismael, M. F., & Thanoon, K. H. (2022). *ResearchGate*. https://doi.org/https://doi.org/10.1109/CCTR1061903.2022.10031608

Kaspersky. (2025). *What is social engineering*?: https://latam.kaspersky.com/resource-center/definitions/what-is-social-engineering

La Vanguardia. (2022). *Technology*. Malicious apps - You should delete these 35 apps from your Android: https://www.lavanguardia.com/tecnologia/20220819/8473935/deberias-eliminar-35-aplicaciones-android-pmv.html

LISA Institute. (2025). *Practical Guide to Social Engineering*: https://www.lisainstitute.com/blogs/blog/guia-practica-ingenieria-social

Malwarebytes. (2025). *What is social engineering?* | How to protect yourself: https://www.malwarebytes.com/es/social-engineering

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743

ESET Awards. (2017). *Machine learning*: Malware analysis and evasion using a heuristic approach: https://premios.eset-la.com/universitario/pdf/machine-learning-analisis-evasion-malware.pdf

Rivera, R. (2014). *Polytechnic University of Madrid*. Analysis of static characteristics of executable files for malware classification: https://oa.upm.es/34343/1/TFM_RICHARD_RIVERA.pdf

Rojo, A. (2020). *University of Valladolid*. Mobile application analysis platform: https://uvadoc.uva.es/bitstream/handle/10324/43269/TFG-G4449.pdf;jsessionid=3AFBD818C74F14396B72DC4CC7A9836E?sequence=1

SECURELIST. (2025). Crimeware and financial cyber threats in 2025: https://securelist.lat/ksb-financial-and-crimeware-predictions-2025/99268/

SECURESOFT. (2022). *News - News, alerts, and reports to help you stay informed*. New Sharkbot Malware Activity: https://www.securesoftcorp.com/w/novedades/ss_alerta693

University of Seville. (2020). *University of Seville*. Dynamic Malware Analysis 2020: https://biblus.us.es/bibing/proyectos/abreproy/94086/fichero/TFG-4086+Aroca+P%C3%A1ez.pdf

UPS. (2025). *Salesian Polytechnic University - Cuenca Campus*. Methodology for malware analysis in a controlled environment: https://dspace.ups.edu.ec/bitstream/123456789/14202/1/UPS-CT006985.pdf

Received July 13, 2025/ Approved September 08, 2025 Pages: 39-55
eISSN: 2600-5743