Trabajo Individual IFO

I. EJERCICIOS:

➢ Mencione un ejemplo de cada tipo de investigación:

- ***En función del propósito -->*** Podríamos mencionar tal vez la investigación de la estructura interna de un átomo, sin motivos de aplicar aquel conocimiento dando uso a tecnologías específicas como un ejemplo de una "investigación teórica".

- ***Por su nivel de profundidad -->*** En este caso podemos mencionar a la investigación de la relación entre el estado del ánimo de una persona y su efectividad en la forma de trabajar, siendo la misma una "investigación correlacional" complementando información de una investigación con la otra.

- ***Por la naturaleza de los datos y la información -->*** Para este caso podríamos mencionar la investigación recopilando datos numéricos de las horas de estudio de cada estudiante, con el fin de analizar su rendimiento académico, siendo la misma una "investigación cuantitativa" al recolectar datos numéricos y siendo una recopilación de datos primarios al crearlos nosotros como tal y no "secundarios" siendo los mismo de otra persona.

- ***Por los medios para obtener los datos -->*** Podríamos considerar una "investigación documental" a la revisión de documentos históricos y registros para el análisis de la salud pública en el país.

- ***Por la mayor o menor manipulación de variables -->*** Se podría considerar una "investigación no experimental" al realizamiento de una encuesta a los estudiantes para saber sus hábitos de estudio, sin intervenir en la búsqueda de variables en su actuar.

- ***Según el tipo de inferencia -->*** En este caso y trabajando con una "inferencia inductiva" podríamos tal vez mencionar que tras observar que la lluvia mojó el pavimento se concluye que la lluvia causa que el mismo se moje.

- ***Según el periodo temporal en que se realiza -->*** En este caso como una "investigación longitudinal" podríamos mencionar el estudio durante años y el comparamiento de los datos del aumento del precio del dólar.

- ***De acuerdo al tiempo en que se efectúan -->*** En el caso de la "investigación sincrónica" podríamos investigar la situación económica actual de cada país.

- Investigue sobre un documento básico de la ética en investigación:

- **La declaración de Belmont:**
  - ✓ Esta declaración establece ciertos principios éticos y directrices con el fin de salvaguardar y proteger a las personas involucradas en la investigación, constando el mismo de tres principios que resultan fundamentales:
    - El respeto a las personas, dirigido al reconocimiento de la autonomía de los individuos y la protección de aquellos con una capacidad reducida de tomar decisiones.
    - La beneficencia con respecto a la maximización de beneficios y la reducción de riesgos a los que participan en la investigación.
    - La justicia llevada al punto de la distribución equitativa de los beneficios de la investigación.

- Investigue sobre una herramienta anti plagio:

- **Plagiarism Checker X:**
  - ✓ Esta es una herramienta de detección de plagio no tan conocida, sin embargo, cuenta con beneficios lo suficientemente buenos como para destacar entre las demás, entre ellos podemos considerar a:
    - Interfaz fácil de usar, siendo la misma intuitiva a gran medida.
    - Compatibilidad de formatos, recibiendo documentos DOCX, PDF, TXT, etc.
    - La velocidad de análisis es regularmente rápida.
    - Cuenta con un sistema que brinda un informe detallado de similitudes encontradas.
    - Finalmente, cuenta con dos funcionalidades, una gratuita, que cuenta con beneficios suficientemente buenos y una de paga, que amplía las herramientas.

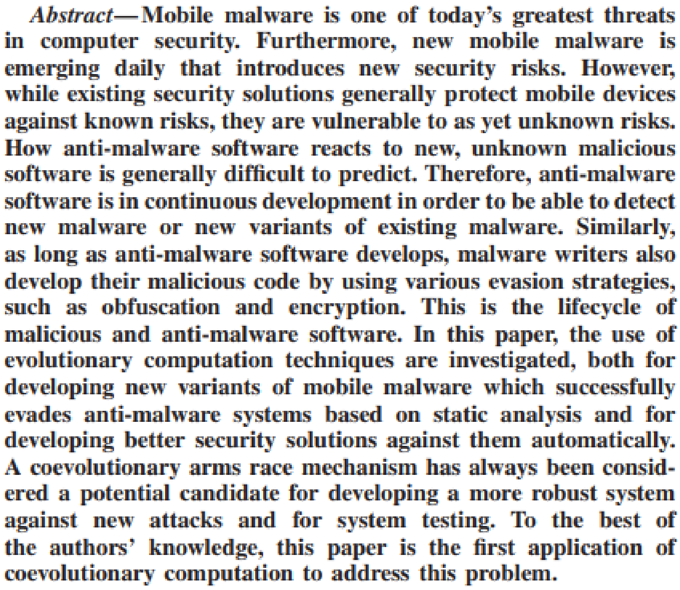
- Realice un análisis de un artículo de investigación en formato IEEE y distinga las diferentes partes que encuentra en el:

- **Artículo a investigar:** ***"Coevolution of Mobile Malware and Anti-Malware"***

  Documento:

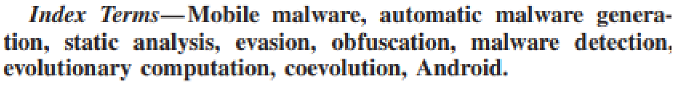
  S. Sen, E. Aydogan and A. I. Aysan, "Coevolution of Mobile Malware and Anti-Malware," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 10, pp. 2563-2574, Oct. 2018, doi: 10.1109/TIFS.2018.2824250.
  keywords: {Malware;Security;Static analysis;Androids;Humanoid robots;Mobile handsets;Mobile malware;automatic malware generation;static analysis;evasion;obfuscation;malware detection;evolutionary computation;coevolution;Android},

✓ Resumen:



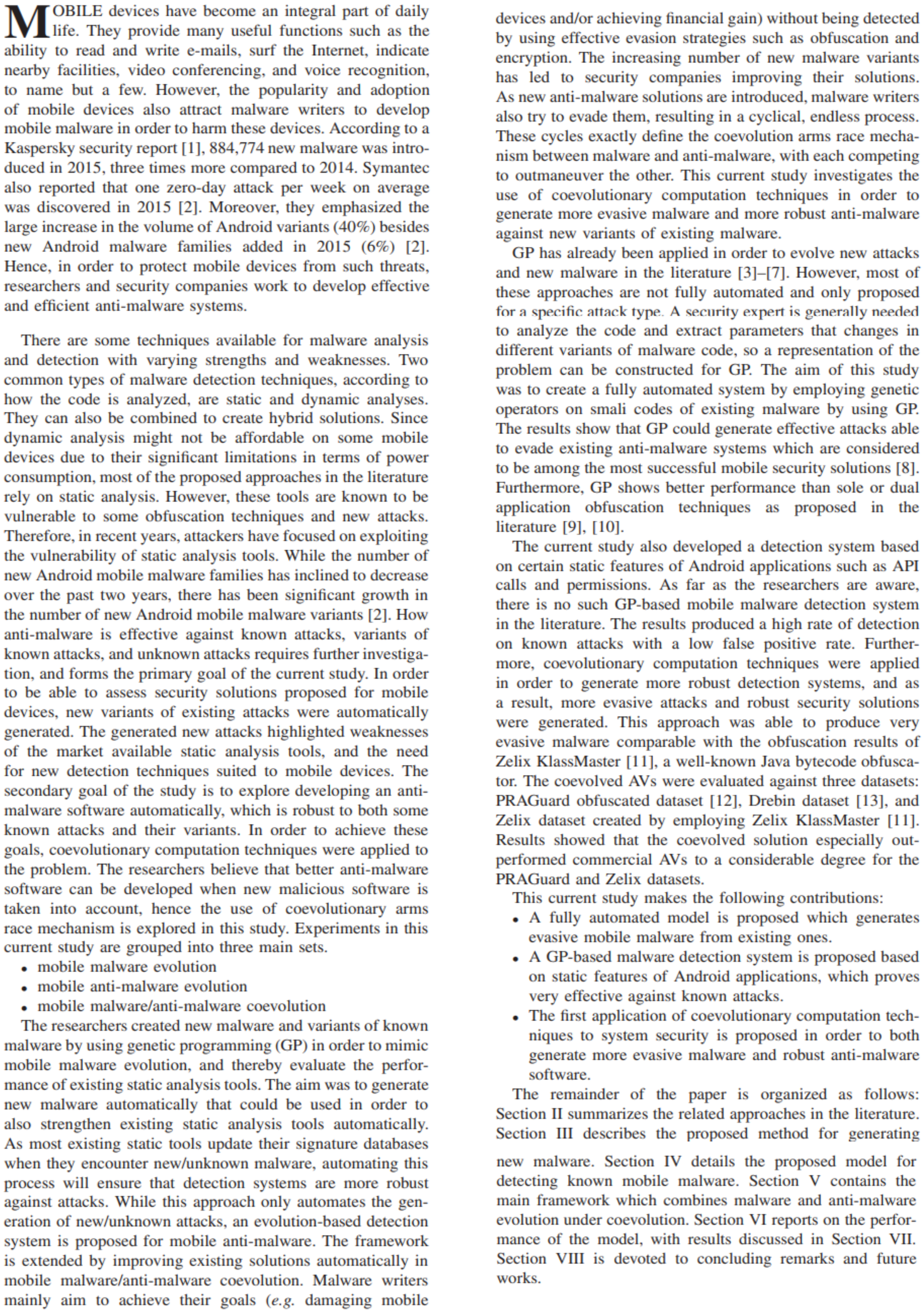
Abstract—Mobile malware is one of today's greatest threats in computer security. Furthermore, new mobile malware is emerging daily that introduces new security risks. However, while existing security solutions generally protect mobile devices against known risks, they are vulnerable to as yet unknown risks. How anti-malware software reacts to new, unknown malicious software is generally difficult to predict. Therefore, anti-malware software is in continuous development in order to be able to detect new malware or new variants of existing malware. Similarly, as long as anti-malware software develops, malware writers also develop their malicious code by using various evasion strategies, such as obfuscation and encryption. This is the lifecycle of malicious and anti-malware software. In this paper, the use of evolutionary computation techniques are investigated, both for developing new variants of mobile malware which successfully evades anti-malware systems based on static analysis and for developing better security solutions against them automatically. A coevolutionary arms race mechanism has always been considered a potential candidate for developing a more robust system against new attacks and for system testing. To the best of the authors' knowledge, this paper is the first application of coevolutionary computation to address this problem.

✓ Palabras clave:



Index Terms—Mobile malware, automatic malware generation, static analysis, evasion, obfuscation, malware detection, evolutionary computation, coevolution, Android.

✓ Introducción:



MOBILE devices have become an integral part of daily life. They provide many useful functions such as the ability to read and write e-mails, surf the Internet, indicate nearby facilities, video conferencing, and voice recognition, to name but a few. However, the popularity and adoption of mobile devices also attract malware writers to develop mobile malware in order to harm these devices. According to a Kaspersky security report [1], 884,774 new malware was introduced in 2015, three times more compared to 2014. Symantec also reported that one zero-day attack per week on average was discovered in 2015 [2]. Moreover, they emphasized the large increase in the volume of Android variants (40%) besides new Android malware families added in 2015 (6%) [2]. Hence, in order to protect mobile devices from such threats, researchers and security companies work to develop effective and efficient anti-malware systems.

There are some techniques available for malware analysis and detection with varying strengths and weaknesses. Two common types of malware detection techniques, according to how the code is analyzed, are static and dynamic analyses. They can also be combined to create hybrid solutions. Since dynamic analysis might not be affordable on some mobile devices due to their significant limitations in terms of power consumption, most of the proposed approaches in the literature rely on static analysis. However, these tools are known to be vulnerable to some obfuscation techniques and new attacks. Therefore, in recent years, attackers have focused on exploiting the vulnerability of static analysis tools. While the number of new Android mobile malware families has inclined to decrease over the past two years, there has been significant growth in the number of new Android mobile malware variants [2]. How anti-malware is effective against known attacks, variants of known attacks, and unknown attacks requires further investigation, and forms the primary goal of the current study. In order to be able to assess security solutions proposed for mobile devices, new variants of existing attacks were automatically generated. The generated new attacks highlighted weaknesses of the market available static analysis tools, and the need for new detection techniques suited to mobile devices. The secondary goal of the study is to explore developing an anti-malware software automatically, which is robust to both some known attacks and their variants. In order to achieve these goals, coevolutionary computation techniques were applied to the problem. The researchers believe that better anti-malware software can be developed when new malicious software is taken into account, hence the use of coevolutionary arms race mechanism is explored in this study. Experiments in this current study are grouped into three main sets.

- mobile malware evolution
- mobile anti-malware evolution
- mobile malware/anti-malware coevolution

The researchers created new malware and variants of known malware by using genetic programming (GP) in order to mimic mobile malware evolution, and thereby evaluate the performance of existing static analysis tools. The aim was to generate new malware automatically that could be used in order to also strengthen existing static analysis tools automatically. As most existing static tools update their signature databases when they encounter new/unknown malware, automating this process will ensure that detection systems are more robust against attacks. While this approach only automates the generation of new/unknown attacks, an evolution-based detection system is proposed for mobile anti-malware. The framework is extended by improving existing solutions automatically in mobile malware/anti-malware coevolution. Malware writers mainly aim to achieve their goals (*e.g.* damaging mobile devices and/or achieving financial gain) without being detected by using effective evasion strategies such as obfuscation and encryption. The increasing number of new malware variants has led to security companies improving their solutions. As new anti-malware solutions are introduced, malware writers also try to evade them, resulting in a cyclical, endless process. These cycles exactly define the coevolution arms race mechanism between malware and anti-malware, with each competing to outmaneuver the other. This current study investigates the use of coevolutionary computation techniques in order to generate more evasive malware and more robust anti-malware against new variants of existing malware.

GP has already been applied in order to evolve new attacks and new malware in the literature [3]–[7]. However, most of these approaches are not fully automated and only proposed for a specific attack type. A security expert is generally needed to analyze the code and extract parameters that changes in different variants of malware code, so a representation of the problem can be constructed for GP. The aim of this study was to create a fully automated system by employing genetic operators on smali codes of existing malware by using GP. The results show that GP could generate effective attacks able to evade existing anti-malware systems which are considered to be among the most successful mobile security solutions [8]. Furthermore, GP shows better performance than sole or dual application obfuscation techniques as proposed in the literature [9], [10].

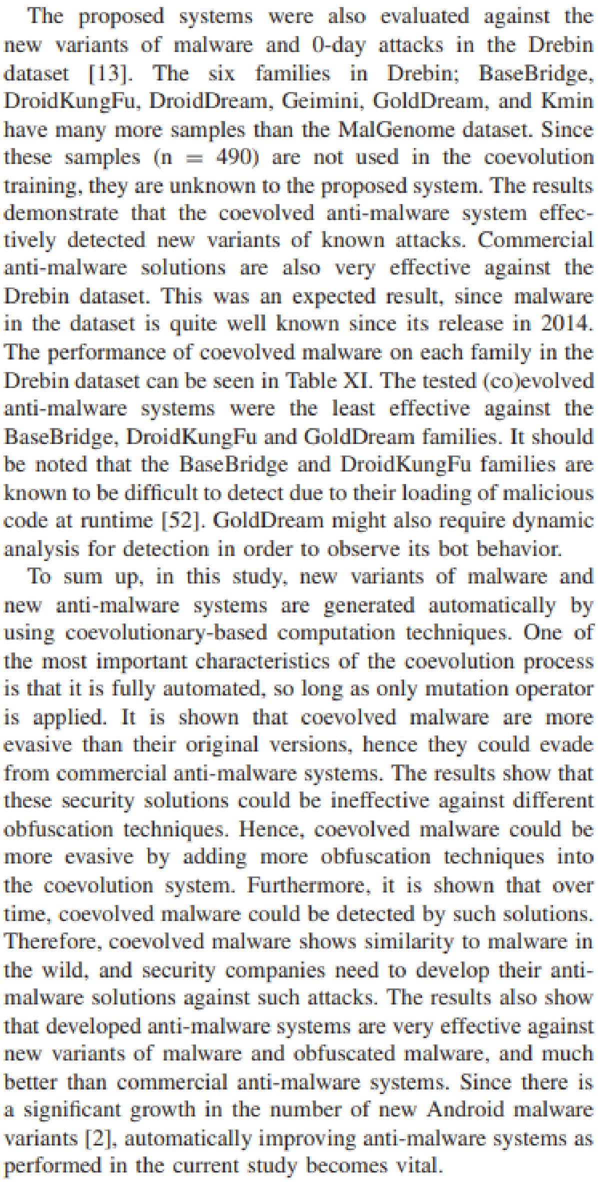
The current study also developed a detection system based on certain static features of Android applications such as API calls and permissions. As far as the researchers are aware, there is no such GP-based mobile malware detection system in the literature. The results produced a high rate of detection on known attacks with a low false positive rate. Furthermore, coevolutionary computation techniques were applied in order to generate more robust detection systems, and as a result, more evasive attacks and robust security solutions were generated. This approach was able to produce very evasive malware comparable with the obfuscation results of Zelix KlassMaster [11], a well-known Java bytecode obfuscator. The coevolved AVs were evaluated against three datasets: PRAGuard obfuscated dataset [12], Drebin dataset [13], and Zelix dataset created by employing Zelix KlassMaster [11]. Results showed that the coevolved solution especially outperformed commercial AVs to a considerable degree for the PRAGuard and Zelix datasets.

This current study makes the following contributions:

- A fully automated model is proposed which generates evasive mobile malware from existing ones.
- A GP-based malware detection system is proposed based on static features of Android applications, which proves very effective against known attacks.
- The first application of coevolutionary computation techniques to system security is proposed in order to both generate more evasive malware and robust anti-malware software.

The remainder of the paper is organized as follows: Section II summarizes the related approaches in the literature. Section III describes the proposed method for generating new malware. Section IV details the proposed model for detecting known mobile malware. Section V contains the main framework which combines malware and anti-malware evolution under coevolution. Section VI reports on the performance of the model, with results discussed in Section VII. Section VIII is devoted to concluding remarks and future works.

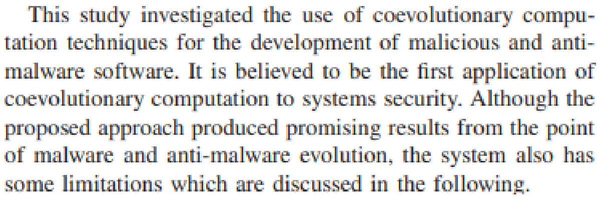
✓ Resultados:



The proposed systems were also evaluated against the new variants of malware and 0-day attacks in the Drebin dataset [13]. The six families in Drebin; BaseBridge, DroidKungFu, DroidDream, Geimini, GoldDream, and Kmin have many more samples than the MalGenome dataset. Since these samples (n = 490) are not used in the coevolution training, they are unknown to the proposed system. The results demonstrate that the coevolved anti-malware system effectively detected new variants of known attacks. Commercial anti-malware solutions are also very effective against the Drebin dataset. This was an expected result, since malware in the dataset is quite well known since its release in 2014. The performance of coevolved malware on each family in the Drebin dataset can be seen in Table XI. The tested (co)evolved anti-malware systems were the least effective against the BaseBridge, DroidKungFu and GoldDream families. It should be noted that the BaseBridge and DroidKungFu families are known to be difficult to detect due to their loading of malicious code at runtime [52]. GoldDream might also require dynamic analysis for detection in order to observe its bot behavior.

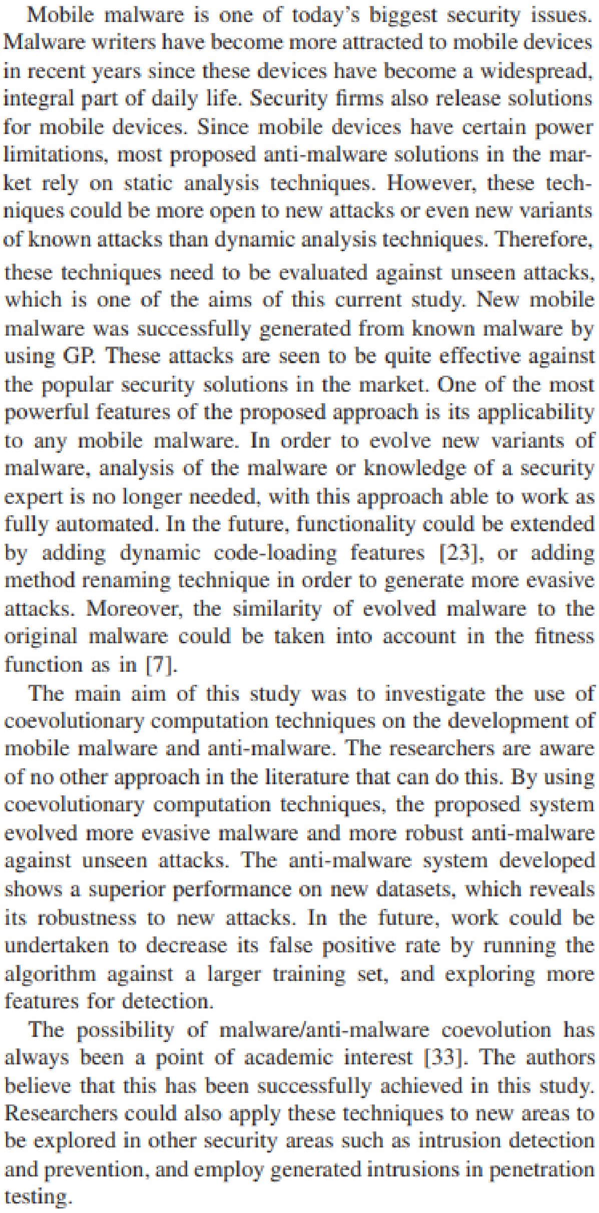
To sum up, in this study, new variants of malware and new anti-malware systems are generated automatically by using coevolutionary-based computation techniques. One of the most important characteristics of the coevolution process is that it is fully automated, so long as only mutation operator is applied. It is shown that coevolved malware are more evasive than their original versions, hence they could evade from commercial anti-malware systems. The results show that these security solutions could be ineffective against different obfuscation techniques. Hence, coevolved malware could be more evasive by adding more obfuscation techniques into the coevolution system. Furthermore, it is shown that over time, coevolved malware could be detected by such solutions. Therefore, coevolved malware shows similarity to malware in the wild, and security companies need to develop their anti-malware solutions against such attacks. The results also show that developed anti-malware systems are very effective against new variants of malware and obfuscated malware, and much better than commercial anti-malware systems. Since there is a significant growth in the number of new Android malware variants [2], automatically improving anti-malware systems as performed in the current study becomes vital.

✓ Discusión:



This study investigated the use of coevolutionary computation techniques for the development of malicious and anti-malware software. It is believed to be the first application of coevolutionary computation to systems security. Although the proposed approach produced promising results from the point of malware and anti-malware evolution, the system also has some limitations which are discussed in the following.

✓ Conclusiones:

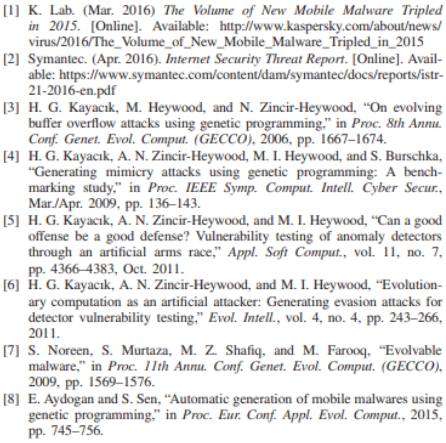


Mobile malware is one of today's biggest security issues. Malware writers have become more attracted to mobile devices in recent years since these devices have become a widespread, integral part of daily life. Security firms also release solutions for mobile devices. Since mobile devices have certain power limitations, most proposed anti-malware solutions in the market rely on static analysis techniques. However, these techniques could be more open to new attacks or even new variants of known attacks than dynamic analysis techniques. Therefore, these techniques need to be evaluated against unseen attacks, which is one of the aims of this current study. New mobile malware was successfully generated from known malware by using GP. These attacks are seen to be quite effective against the popular security solutions in the market. One of the most powerful features of the proposed approach is its applicability to any mobile malware. In order to evolve new variants of malware, analysis of the malware or knowledge of a security expert is no longer needed, with this approach able to work as fully automated. In the future, functionality could be extended by adding dynamic code-loading features [23], or adding method renaming technique in order to generate more evasive attacks. Moreover, the similarity of evolved malware to the original malware could be taken into account in the fitness function as in [7].

The main aim of this study was to investigate the use of coevolutionary computation techniques on the development of mobile malware and anti-malware. The researchers are aware of no other approach in the literature that can do this. By using coevolutionary computation techniques, the proposed system evolved more evasive malware and more robust anti-malware against unseen attacks. The anti-malware system developed shows a superior performance on new datasets, which reveals its robustness to new attacks. In the future, work could be undertaken to decrease its false positive rate by running the algorithm against a larger training set, and exploring more features for detection.

The possibility of malware/anti-malware coevolution has always been a point of academic interest [33]. The authors believe that this has been successfully achieved in this study. Researchers could also apply these techniques to new areas to be explored in other security areas such as intrusion detection and prevention, and employ generated intrusions in penetration testing.

✓ Referencias:



[1] K. Lab. (Mar. 2016) *The Volume of New Mobile Malware Tripled in 2015*. [Online]. Available: http://www.kaspersky.com/about/news/virus/2016/The_Volume_of_New_Mobile_Malware_Tripled_in_2015

[2] Symantec. (Apr. 2016). *Internet Security Threat Report*. [Online]. Available: https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf

[3] H. G. Kayacık, M. Heywood, and N. Zincir-Heywood, "On evolving buffer overflow attacks using genetic programming," in *Proc. 8th Annu. Conf. Genet. Evol. Comput. (GECCO)*, 2006, pp. 1667–1674.

[4] H. G. Kayacık, A. N. Zincir-Heywood, M. I. Heywood, and S. Burschka, "Generating mimicry attacks using genetic programming: A benchmarking study," in *Proc. IEEE Symp. Comput. Intell. Cyber Secur.*, Mar./Apr. 2009, pp. 136–143.

[5] H. G. Kayacık, A. N. Zincir-Heywood, and M. I. Heywood, "Can a good offense be a good defense? Vulnerability testing of anomaly detectors through an artificial arms race," *Appl. Soft Comput.*, vol. 11, no. 7, pp. 4366–4383, Oct. 2011.

[6] H. G. Kayacık, A. N. Zincir-Heywood, and M. I. Heywood, "Evolutionary computation as an artificial attacker: Generating evasion attacks for detector vulnerability testing," *Evol. Intell.*, vol. 4, no. 4, pp. 243–266, 2011.

[7] S. Noreen, S. Murtaza, M. Z. Shafiq, and M. Farooq, "Evolvable malware," in *Proc. 11th Annu. Conf. Genet. Evol. Comput. (GECCO)*, 2009, pp. 1569–1576.

[8] E. Aydogan and S. Sen, "Automatic generation of mobile malwares using genetic programming," in *Proc. Eur. Conf. Appl. Evol. Comput.*, 2015, pp. 745–756.