UNIVERSITI PUTRA MALAYSIA

A BEHAVIOUR-BASED ANALYTICAL MALWARE DETECTION FRAMEWORK FOR ANDROID SMARTPHONES

MOHSEN DAMSHENAS

FSKTM 2014 24

A BEHAVIOUR-BASED ANALYTICAL MALWARE DETECTION FRAMEWORK FOR ANDROID SMARTPHONES

By

MOHSEN DAMSHENAS

© COPYRIGHT UPM

Thesis Submitted to the School of Graduate Studies, University Putra Malaysia, in Fulfilment of the Requirements for the Degree of Master of Science

October 2014

COPYRIGHT

All material contained within the thesis, including without limitation text, logos, icons, photographs and all other artwork, is copyright material of Universiti Putra Malaysia unless otherwise stated. Use may be made of any material contained within the thesis for non-commercial purposes from the copyright holder. Commercial use of material may only be made with the express, prior, written permission of Universiti Putra Malaysia.

Copyright © Universiti Putra Malaysia

© COPYRIGHT UPM

This thesis is dedicated to my parents.

For their endless love, support and encouragement

© COPYRIGHT UPM

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Master of Science

A BEHAVIOUR-BASED ANALYTICAL MALWARE DETECTION FRAMEWORK FOR ANDROID SMARTPHONES

By

MOHSEN DAMSHENAS

October 2014

Chairman: Ali Dehghantanha, PhD

Faculty: Computer Science and Information Technology

The fast growth in the number of Android smartphone users and the lack of suitable malware detection techniques for these devices attract vicious minds to infect users with malicious software. The fact is that today, after more than seven years of initial Android release, there are still malwares spreading in official Android markets. It is necessary to mention that not only the number of users are being increased, the user's data becoming more and more sensitive. Nowadays, a typical smartphone can contain contact information, private messages, location information, emails or even credit card numbers. Previous studies reported that the initial detection rate of a newly created Android virus is less than 5%, which indicate that the available products in the market are not really effective. Considering the sharp increase in number of mobile malwares and the ineffectiveness of current malware detection solutions, Android users are facing a great problem.

In this research, we propose a behaviour-based analytical malware detection framework for Android smartphones (which in known as Nestor). This framework has three main models. The first model is in charge of keeping the primary dataset up to dated. Then the analyser model, M0Droid, utilises behaviour-based malware detection approach to obtain the behavioural factors and generate a signature for every application. This signature is generated based on the system call requests by application and then normalised with median and z-score for generating more accurate and effective signature. It then uses Spearman's rank correlation coefficient to identify similar malware signatures in a previously generated blacklist of malwares signature. The result of all these processing appears in a safe Android market that the end user can download Android application without worrying about malware infection.

The outcome of the M0Droid accuracy measurement experiment against malware dataset indicates 60.16% positives malware detection, 39.43% false-positives and 0.4% false-negatives with choosing Spearman correlation coefficient rank of 0.90 as the threshold. This threshold is directly proportional to the false-negative rate while it is inversely proportional to positive and false-positive rates.

© COPYRIGHT UPM

Moreover, to compare our result with a similar model, we employed the same evaluation method as Crowdroid used to test M0Droid. The result represents an improvement in detection rate since Crowdroid were able to detect 97% of malwares while M0Droid detect all malwares in test environment.

It is notable, that the novelty of this work and the most effective factors in obtaining these results are due to employing Linux Monkey for mimicking the user input, z-score for signature normalisation and Spearman's rank correlation coefficient for signatures comparison. We hope this research can be a stepping stone for improvement in Android malware detection techniques and development of safe Android markets which eventually increase the security of end-user devices.

© COPYRIGHT UPM

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia Sebagai memenuhi keperluan untuk ijazah Master Sains

A BEHAVIOUR-BASED ANALYTICAL MALWARE DETECTION FRAMEWORK FOR ANDROID SMARTPHONES

Oleh

MOHSEN DAMSHENAS

Oktober 2014

Pengerusi: Ali Dehghantanha, PhD

Fakulti: Sains Komputer dan Teknologi Maklumat

Bilangan pengguna telefon pintar *Android* yang semakin meningkat dan kekurangan teknik mengesan *malware* yang sesuai untuk peranti ini telah menarik minda-minda jahat untuk menyebarkan perisian berbahaya kepada pengguna. Hakikatnya setelah lebih tujuh tahun pengeluaran *Android* bermula, masih ada *malware* yang menular di pasaran rasmi *Android*. Andalah penting untuk dinyatakan bahawa bukan sahaja bilangan pengguna telah meningkat malah data peribadi pengguna juga semakin sensitif. Dewasa ini, telefon pintar biasa mengandungi maklumat perhubungan, mesej peribadi, maklumat lokasi, e-mel malahan nombor kad kredit.

Kajian sebelum ini menunjukkan kadar pengesanan awal bagi *virus Android* yang terbaru adalah kurang daripada 5% di mana ini menunjukkan produk yang sedia ada di pasaran adalah kurang berkesan. Mempertimbangkan penambahan bilangan *malware* peranti mudah alih yang semakin galak dan ketidakberkesanan penyelesaian pengesan malware yang ada, pengguna *Android* sedang menghadapi masalah yang serius. Dalam kajian ini, kami mencadangkan rangka kerja berasaskan analisis tingkah laku pengesan *malware* untuk telefon pintar *Android* (dikenali sebagai *Nestor*).

Rangka kerja ini mengandungi tiga model utama. Model yang pertama bertanggungjawab untuk mengekalkan set data utama sentiasa di kemaskini. Seterusnya model analisis, *M0Droid*, menggunakan pendekatan pengesan *malware* berasaskan tingkah laku untuk mendapatkan faktor perlakuan dan menghasilkan identifikasi untuk setiap aplikasi. Identifikasi ini dihasilkan berasaskan permintaan sistem panggilan dari aplikasi dan kemudian dinormalkan dengan *median* dan z-skor untuk menjanakan identifikasi yang lebih cepat dan berkesan

Ia kemudiannya menggunakan pekali kolerasi kedudukan *Spearman* untuk mengenal pasti identifikasi *malware* yang sama dalam senarai hitam identifikasi *malware* yang dihasilkan sebelum ini. Hasil daripada semua pengolahan ini dapat dilihat dalam pasaran *Android* yang selamat di mana pengguna boleh memuat turun aplikasi *Android* tanpa khuatir tentang jangkitan *malware*. Hasil kajian ketepatan ukuran *M0Droid* terhadap set data *malware* menunjukkan 60.16% positif pengesanan

© COPYRIGHT UPM

malware, 39.43% palsu-positif dan 0.4% palsu-negatif dengan memilih pekali kolerasi *Spearman* pada kedudukan 0.9 sebagai ambang.

Ambang ini berkadar terus dengan kadar palsu-negatif sementara ianya berkadar songsang dengan kadar positif dan palsu-positif. Di samping itu, untuk membandingkan hasil kajian kami dengan model yang hampir sama, kami menerapkan kaedah penilaian yang sama seperti *Crowdroid* gunakan untuk menguji *M0Droid*. Hasil kajian ini menunjukkan peningkatan dalam kadar pengesanan sejak *Crowdroid* berjaya mengesan 97% malware sementara *M0Droid* mengesan semua *malware* dalam ujian persekitaran.

Ianya ketara bahawa pembaharuan kerja ini dan faktor yang paling berkesan dalam mendapatkan keputusan adalah dengan menggunakan *Linux Monkey* untuk meniru input pengguna, z-skor untuk penormalan identifikasi dan pekali kolerasi kedudukan *Spearman* sebagai perbandingan identifikasi. Kami berharap kajian ini dapat menjadi batu loncatan kepada peningkatan teknik untuk mengesan malware di dalam *Android* dan juga perkembangan pasaran *Android* yang selamat dan akhirnya meningkatkan tahap keselamatan peranti pengguna.

© COPYRIGHT UPM

ACKNOWLEDGEMENTS

It is a great opportunity to thank Dr Ali Dehghantanha, Professor Dr Ramlan Mahmod, and Dr Solahuddin bin Shamsuddin for their great help on this thesis and for their supporting guidance, ideas, and materials. And to Faculty of Computer Science and Information Technology for supporting my research.

Finally, my special thanks belong to my family who support my decisions with their motivations and moral support I needed to complete this thesis; and to my friend and colleagues for their kind support and advice.

© COPYRIGHT UPM

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Master of Science. The members of the Supervisory Committee were as follows:

Ali Dehghantanha, Ph.D.

Senior Lecturer

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

(Chairman)

Ramlan Mahmod, Ph.D.

Professor

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

(Member)

Solahuddin Bin Shamsuddin, Ph.D.

Chief Technology Officer

CyberSecurity Malaysia

(External Member)

BUJANG BIN KIM HUAT, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia
Date:

© COPYRIGHT UPM

Declaration by Graduate Student

I hereby confirm that:

- this thesis is my original work;
- quotations, illustrations and citations have been duly referenced;
- this thesis has not been submitted previously or concurrently for any other degree at any other institutions;
- intellectual property from the thesis and copyright of thesis are fully-owned by Universiti Putra Malaysia, as according to the Universiti Putra Malaysia (Research) Rules 2012;
- written permission must be obtained from supervisor and the office of Deputy Vice-Chancellor (Research and Innovation) before thesis is published (in the form of written, printed or in electronic form) including books, journals, modules, proceedings, popular writings, seminar papers, manuscripts, posters, reports, lecture notes, learning modules or any other materials as stated in the Universiti Putra Malaysia (Research) Rules 2012;
- there is no plagiarism or data falsification/fabrication in the thesis, and scholarly integrity is upheld as according to the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) and the Universiti Putra Malaysia (Research) Rules 2012. The thesis has undergone plagiarism detection software.

Signature: Date:

Name and Matric No.: Mohsen Damshenas (GS33495)

© COPYRIGHT UPM

Declaration by Members of Supervisory Committee

This is to confirm that:

- the research conducted and the writing of this thesis was under our supervision;
- supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) are adhered to.

Signature:	----------	Signature:	----------
Name of Chairman of Supervisory Committee:	Ali Dehghantanha	Name of Member of Supervisory Committee:	Ramlan Mahmod
Signature:	----------		
Name of Member of Supervisory Committee:	Solahuddin Bin Shamsuddin		

© COPYRIGHT UPM

TABLE OF CONTENTS

© COPYRIGHT UPM

© COPYRIGHT UPM

LIST OF TABLES

© COPYRIGHT UPM

LIST OF FIGURES

© COPYRIGHT UPM

LIST OF ABBREVIATIONS

AAPT	Android Asset Packing Tool
ADB	Android Debugger Bridge
API	Application Program Interface
APP	Application
AVD	Android Virtual Device
BOS	Behaviour Operation Set
BYOD	Bring Your Own Device
FN	False Negative
FP	False Positive
GLIBC	GNU C Library
GOODWARE	Good Software
IDS	Intrusion Detection System
KBTA	Knowledge Based Temporal Abstraction
MALWARE	Malicious Software
MBF	Malicious Behaviour Feature
MIDM	Man In The Middle
P	Positive
PC	Personal Computer
SVM	Support Vector Machine
SysCall	System Call

© COPYRIGHT UPM

CHAPTER 1

INTRODUCTION

The fast growth in the number of shipped smartphones raised the number of attacks to users through different techniques. According to the prediction of Juniper networks annual report, it is estimated that 1 billion Android based smart phones will be distributed in 2017 ("Juniper Networks Third Annual Mobile Threats Report," 2013). Besides, the improvements in technology of smartphone brought huge processing power and vast amount of storages; this makes the smartphone capable of handling and containing ultra-sensitive information (i.e. banking information or emails). The incredible usefulness of smartphones made them a primary target for vicious minds and Android as a leading platform in smartphone operating systems, attracts a huge load of malicious activities.

One of the major concerns of Android smartphones security is malware (malicious software) infection as it facilitates automated attack to millions of users with minimum supervision (Vidas et al., 2011) The penetration of malicious software into the official app (application) markets and utilising the most complicated approaches to evade anti-malware solutions made this problem a complicated confound. To make the matter worse, Imperva assessment of antivirus effectiveness reports that "The initial detections rate of a newly created virus is less than 5%" (Imperva, 2012) which indicate that the available products in the market are not really effective. The fact is that today, after more than seven years of Android initial release, there are still malwares spreading in Android official markets. The third annual Mobile Threat Report from Juniper Networks indicates that a 614% increase of mobile malicious software growth from March 2012 to March 2013 ("Juniper Networks Third Annual Mobile Threats Report," 2013).

In this research, we propose a behaviour-based analytical malware detection framework for Android smartphones. This framework consists of a model for collecting and classifying a malware dataset in the first place, a model for measuring the accuracy of the malware detection method, and a model for offering a safe Android market to the end user.

This model comprise of different methods such as the main method for identifying the behavioural attributes of any given app (M0Droid), a method for gathering malwares from available dataset, a method for collecting malwares from users' submissions, a method to collect goodware (good software) from Google play (Android official market), a method for processing the safe market app submission, and finally a method for user access to the safe market.

We begin with a theoretical explanation of our framework and included models and techniques; and then provide details of implementing the framework in a test environment. Finally, the usefulness and accuracy of our framework are shown using real world malwares and goodwares.

© COPYRIGHT UPM

In this chapter, we first describe the research gap we aim to fill and the outcomes of this research. Later we discuss the scope of this research and finally we explain the arrangement of this thesis.

1.1. Problem Statement

It is a known fact that popular technologies draw the attention of malicious minds to take advantage of the end users (Mansfield-Devine, 2012). Android smartphone as one of the main trends is not an exception to this fact. Even though the academic community is showing more interested to Android malware detection techniques (Burguera et al., 2011; A. D. Schmidt, Bye, et al., 2009; Shabtai et al., 2012; Zhou & Jiang, 2012), there is still room for more research.

Reports indicate that the number of unique malwares for Android increased from 47% in 2011 to 92% in 2013 ("Juniper Networks Third Annual Mobile Threats Report," 2013). This rapid increase shows this matter is pretty hazardous for the Android platform. Figure 1 shows the growth in share of Android malware samples in comparison to other platforms from December 2011 to March 2013.

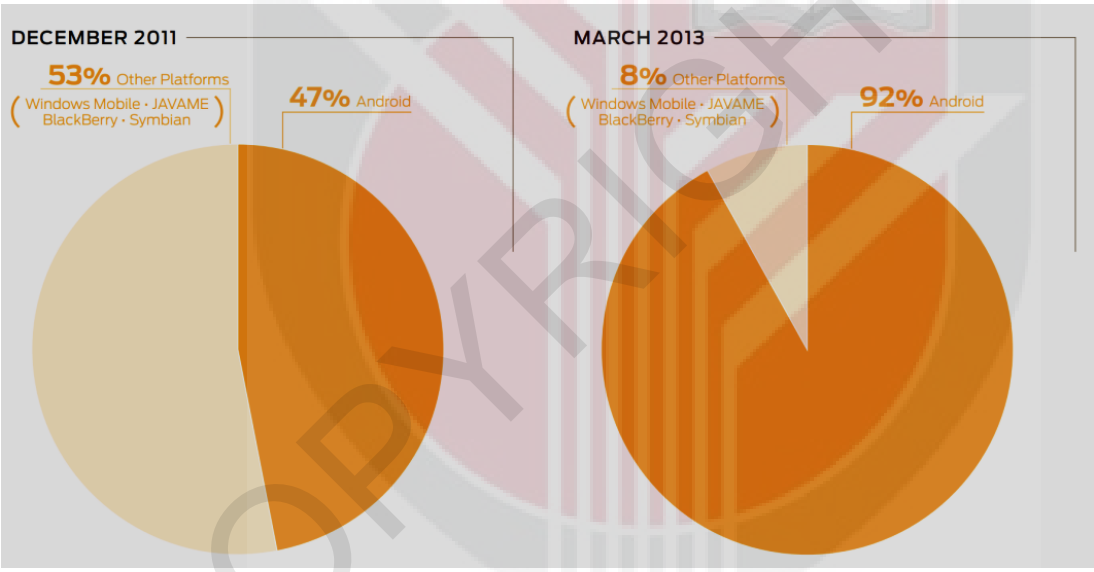

Figure 1 – A comparison of unique malware sample from 2011 to 2013

Preparing an effective solution for defeating the discussed wave of malware is absolutely a necessity with the mind blowing growth of Android users. This solution should not put the end user at risk of infection. Thus from a strategic point of view the malwares should be filtered right from the source (app stores). Researches result show regular scanning of Android app stores, but these frequent scans seem to be ineffective due to a lack of helpful malware detection technique (Enck et al., 2011).

Crowdroid is one of the available solutions for Android malware detection with 97% detection ratio (Burguera et al., 2011). This novel approach collects system call requests of Android applications and forms a signature which can be used for comparison with a blacklist. The main problem with this solution is that it can not

© COPYRIGHT UPM

integrated with Android markets and supports only the end user device. On the other hand, the 97% detection ratio is not enough for Android market because the users need to be sure that the Android market provides only benign applications. In addition, Crowdroid employs server base architecture without securing the communications which makes it vulnerable to man in the middle attacks.

In this research, our proposed framework is able to solve these problems by providing a model for malware analysis, a model for maintaining the malware analyser sources and a model for benefiting the user through a safe Android market.

1.2. Research Objectives

The proposed framework as a solution to the insecurity of the Android app markets fills the gap between behaviour-based malware detection on the server and the end user Android Smartphone. Therefore our research objectives are as follow:

1. To propose a model for collecting and normalising malware samples obtained from different sources. Once these malwares' character is confirmed, they will be used for updating a signature blacklist.
2. To propose an effective behaviour-based malware analysis model for Android smartphones.
3. To propose a safe app store model for delivering reliable apps to the end users. The app store will be accessible through website or the mobile app as well.

1.3. Research Scope

Our research is limited to design and implement a framework for detecting malwares in Android smartphones. It is important to highlight that the framework only supports Android applications and does not support other smartphone platforms. Additionally, as this framework is based on official Android emulator on a server, the performance of the system is completely influenced by the processing power of the server.

The framework supports BYOD (Bring Your Own Device) environments where the organisational policies restricted downloading and installing app from any unauthorised app store. Mainly because the framework security strategy is to provide a safe source for app downloads; therefore the user should not be allowed to download applications from any other app store.

1.4. Research Contributions

The key point for assessing the usefulness of a research is its contributions to the body of knowledge. Pursuing the primary objectives of this research results in proposing a behaviour-based malware detection. This framework is based on a server (cloud) that removes the main load of malware analysis from users' side. The main contributions of this research are as follow:

© COPYRIGHT UPM

1. A model for collecting and characterising malware samples. As this framework relies on having a blacklist, made by unique behavioural signature of the malwares, we needed to design a model for characterising collected malwares and generating a database of malware signatures. This database should remain updated by collecting and analysing new malwares continuously.

2. A model for analysing malwares with lowest false negative rate possible. We devised M0Droid and implement it using Python scripts. Behaviour based malware detection is proven to be more effective than static malware detection. Unlike static malware detection techniques that uses static attributes of the file, behaviour-based techniques use behavioural attributes such as requests for reading files or network access.

3. A model for delivering reliable apps to the end users. As the final phase of our framework, it was required to deliver the final goods to end users and help them benefit from the system. This model ensures that the users have access to reliable and safe applications through our secure Android app store.

1.5. Thesis Organisation

In this research, we study previous researches and related works in Chapter 2. This chapter contains previous related studies on malware analysis, malware propagation, malware types and smartphone malwares. The main objective of this chapter is to obtain a comprehensive view on the area of the research, the problems researchers are dealing with, and similar solutions to what we are about to propose in other chapters of this research.

Chapter 3 demonstrates details of how we planned this research. This chapter concerns the research methodology we used for studying the literature of the area and discover the challenges of the area, designing the main framework to conquer the challenges, developing a system for the designed framework, evaluating the implemented framework, refining the weaknesses of the framework and finally documenting the results.

Next, we propose our solution in Chapter 4. This chapter includes the details of the three main models of the framework and methods used to build it. M0Droid, as the core model of the framework is explained all the way through malware collection and characterisation model, malware analysing model and our Android safe market. It is notable that technical details provided in this section might require precise study of the literature review and having background of mathematics and statistics.

In Chapter 5, we demonstrate the evaluation plans for testing the implemented framework. We explain what variables we used to make the required environment for developing and examining our framework. Basically, there are three different test for evaluating the implemented framework: Testing the accuracy to figure out the statistics of detecting malwares, missing malwares or false detections. Testing compatibility to discover whether the framework can work on different Android

© COPYRIGHT UPM

versions. Testing the scalability to find out whether the framework can produce the same accuracy statistics under a heavy load or not.

At the end, we conclude the research in Chapter 6. In this chapter, an overview of the research is followed by getting to the point where the reader can decide whether the outcomes of the research are achieved or not. In Chapter 7 we argue possible future works. We discuss the possible future works for next generation of researchers, hoping that it may become a motivation for a greater research and further knowledge generation.

1.6. Summary

In this section we provide an introduction to our research. We discuss the problem that motivate this research and then describe our objectives and contributions. The scope of this research is also defined in this section. At the end we present the organisation of this research as literature review, research methodology, Nestor framework, experimental results, conclusion and future works.

© COPYRIGHT UPM

BIBLIOGRAPHY

Bai, G., Gu, L., Feng, T., Guo, Y., & Chen, X. (2010). Context-aware usage control for android. In *Security and Privacy in Communication Networks* (pp. 326-343). Springer Berlin Heidelberg.

Barras, C., & Gauvain, J. L. (2003, April). Feature and score normalization for speaker verification of cellular data. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on* (Vol. 2, pp. II-49). IEEE.

Bayer, U., Kirda, E., & Kruegel, C. (2010, March). Improving the efficiency of dynamic malware analysis. In *Proceedings of the 2010 ACM Symposium on Applied Computing* (pp. 1871-1878). ACM.

Blasing, T., Batyuk, L., Schmidt, A. D., Camtepe, S. A., & Albayrak, S. (2010, October). An android application sandbox system for suspicious software detection. In *Malicious and unwanted software (MALWARE), 2010 5th international conference on* (pp. 55-62). IEEE.

Bose, A., Hu, X., Shin, K. G., & Park, T. (2008, June). Behavioral detection of malware on mobile handsets. In *Proceedings of the 6th international conference on Mobile systems, applications, and services* (pp. 225-238). ACM.

Bowman, I. T., Holt, R. C., & Brewster, N. V. (1999, May). Linux as a case study: Its extracted software architecture. In *Proceedings of the 21st international conference on Software engineering* (pp. 555-563). ACM.

Burguera, I., Zurutuza, U., & Nadjm-Tehrani, S. (2011, October). Crowdroid: behavior-based malware detection system for android. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices* (pp. 15-26). ACM.

Di Cerbo, F., Girardello, A., Michahelles, F., & Voronkova, S. (2011). Detection of malicious applications on android os. In *Computational Forensics* (pp. 138-149). Springer Berlin Heidelberg.

Chandramohan, M., & Tan, H. B. K. (2012). Detection of mobile malware in the wild. *Computer*, *45*(9), 0065-71.

Cheng, J., Wong, S. H., Yang, H., & Lu, S. (2007, June). Smartsiren: virus detection and alert for smartphones. In *Proceedings of the 5th international conference on Mobile systems, applications and services* (pp. 258-271). ACM.

Christodorescu, M., & Jha, S. (2006). Static analysis of executables to detect malicious patterns. Wisconsin Univ-Madison Dept Of Computer Sciences.

Christodorescu, M., Jha, S., & Kruegel, C. (2008, February). Mining specifications of malicious behavior. In *Proceedings of the 1st India software engineering conference* (pp. 5-14). ACM.

Dagon, D., Martin, T., & Starner, T. (2004). Mobile phones as computing devices: The viruses are coming!. *Pervasive Computing, IEEE*, *3*(4), 11-15.

© COPYRIGHT UPM

Enck, W., Gilbert, P., Chun, B. G., Cox, L. P., Jung, J., McDaniel, P., & Sheth, A. N. (2014). TaintDroid: an information flow tracking system for real-time privacy monitoring on smartphones. *Communications of the ACM*, *57*(3), 99-106.

Enck, W., Octeau, D., McDaniel, P., & Chaudhuri, S. (2011, August). A Study of Android Application Security. In *USENIX security symposium* (Vol. 2, p. 2).

Field, A. (2009). Discovering statistics using SPSS. Sage publications.

Gascon, H., Yamaguchi, F., Arp, D., & Rieck, K. (2013, November). Structural detection of android malware using embedded call graphs. In *Proceedings of the 2013 ACM workshop on Artificial intelligence and security* (pp. 45-54). ACM.

Gautheir, T. D. (2001). Detecting trends using spearman's rank correlation coefficient. *Environmental forensics*, *2*(4), 359-362.

Heiman, G. (2013). Basic statistics for the behavioral sciences. Cengage Learning.

Hu, C., & Neamtiu, I. (2011, May). Automating GUI testing for Android applications. In *Proceedings of the 6th International Workshop on Automation of Software Test* (pp. 77-83). ACM.

Hypponen, M. (2006). Malware goes mobile. *Scientific American*, *295*(5), 70-77.

Imgraben, J., Engelbrecht, A., & Choo, K.-K. R. (2014). Always connected, but are smart mobile users getting more security savvy? A survey of smart mobile device users. *Behaviour & Information Technology*, 14(3), 1-14.

Imperva (2012). Assessing the Effectiveness of Antivirus Solutions. Retrieved from www.imperva.com/download.asp?id=324

Jacoby, G. A., Marchany, R., & Davis IV, N. J. (2004, June). Battery-based intrusion detection a first line of defense. In *Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC* (pp. 272-279). IEEE.

Jain, A., Nandakumar, K., & Ross, A. (2005). Score normalization in multimodal biometric systems. *Pattern recognition*, *38*(12), 2270-2285.

Jiang, Q., Zhao, X., & Huang, K. (2011, June). A feature selection method for malware detection. In *Information and Automation (ICIA), 2011 IEEE International Conference on* (pp. 890-895). IEEE.

Juniper Networks, Inc. (2013). Juniper Networks Third Annual Mobile Threats Report. Retrieved from http://www.juniper.net/us/en/local/pdf/additional-resources/jnpr-2012-mobile-threats-report.pdf

Kim, H., Shin, K. G., & Pillai, P. (2011). MODELZ: monitoring, detection, and analysis of energy-greedy anomalies in mobile handsets. *Mobile Computing, IEEE Transactions on*, *10*(7), 968-981.

Lee, T., & Mody, J. J. (2006, April). Behavioral classification. In *EICAR Conference* (pp. 1-17).

Mansfield-Devine, S. (2012). Paranoid Android: just how insecure is the most popular mobile platform?. *Network Security*, *2012*(9), 5-10.

© COPYRIGHT UPM

Martínez, C. A., Echeverri, G. I., & Sanz, A. G. C. (2010, September). Malware detection based on cloud computing integrating intrusion ontology representation. In *Communications (LATINCOM), 2010 IEEE Latin-American Conference on* (pp. 1-6). IEEE.

Marx, M. L., & Larsen, R. J. (2006). Introduction to mathematical statistics and its applications. Pearson/Prentice Hall.

McGrath, R., & Akkerman, W. (2004). Source Forge Strace Project. Retrieved from http://sourceforge.net/projects/strace/

Meier, R. (2012). Professional Android 4 application development. John Wiley & Sons.

Oberheide, J., Bailey, M., & Jahanian, F. (2009, August). PolyPack: an automated online packing service for optimal antivirus evasion. In *Proceedings of the 3rd USENIX conference on Offensive technologies* (pp. 9-9). USENIX Association.

Ongtang, M., McLaughlin, S., Enck, W., & McDaniel, P. (2012). Semantically rich application‐centric security in Android. *Security and Communication Networks*, *5*(6), 658-673.

Park, J. H., Kim, M., Noh, B. N., & Joshi, J. B. (2006, September). A Similarity based Technique for Detecting Malicious Executable files for Computer Forensics. In *Information Reuse and Integration, 2006 IEEE International Conference on* (pp. 188-193). IEEE.

Peck, R., Olsen, C., & Devore, J. (2008). Introduction to statistics and data analysis. Cengage Learning.

Pirie, W. (1988). Spearman rank correlation coefficient. *Encyclopedia of statistical sciences*.

Portokalidis, G., Homburg, P., Anagnostakis, K., & Bos, H. (2010, December). Paranoid Android: versatile protection for smartphones. In *Proceedings of the 26th Annual Computer Security Applications Conference* (pp. 347-356). ACM.

Provos, N. (2003, August). Improving Host Security with System Call Policies. In *USENIX Security* (Vol. 3).

Ramilli, M., Bishop, M., & Sun, S. (2011, October). Multiprocess malware. In Malicious and Unwanted Software (MALWARE), 2011 6th International Conference on (pp. 8-13). IEEE.

Ramilli, M., & Prandini, M. (2010). Always the Same, Never the Same. *Security & Privacy, IEEE*, *8*(2), 73-75.

Rieck, K., Holz, T., Willems, C., Düssel, P., & Laskov, P. (2008). Learning and classification of malware behavior. In *Detection of Intrusions and Malware, and Vulnerability Assessment* (pp. 108-125). Springer Berlin Heidelberg.

Schmidt, A. D., Camtepe, S. A., & Albayrak, S. (2010). Static smartphone malware detection.

© COPYRIGHT UPM

Schmidt, A. D., Bye, R., Schmidt, H. G., Clausen, J., Kiraz, O., Yuksel, K. A., ... & Albayrak, S. (2009, June). Static analysis of executables for collaborative malware detection on android. In *Communications, 2009. ICC'09. IEEE International Conference on* (pp. 1-5). IEEE.

Schmidt, A. D., Clausen, J. H., Camtepe, A., & Albayrak, S. (2009, October). Detecting symbian os malware through static function call analysis. In *Malicious and Unwanted Software (MALWARE), 2009 4th International Conference on* (pp. 15-22). IEEE.

Shabtai, A., Fledel, Y., Kanonov, U., Elovici, Y., Dolev, S., & Glezer, C. (2010). Google android: A comprehensive security assessment. *IEEE Security and Privacy*, *8*(2), 35-44.

Shabtai, A., Kanonov, U., & Elovici, Y. (2010). Intrusion detection for mobile devices using the knowledge-based, temporal abstraction method. *Journal of Systems and Software*, *83*(8), 1524-1537.

Shabtai, A., Kanonov, U., Elovici, Y., Glezer, C., & Weiss, Y. (2012). "Andromaly": a behavioral malware detection framework for android devices. *Journal of Intelligent Information Systems*, *38*(1), 161-190.

Shabtai, A., Moskovitch, R., Elovici, Y., & Glezer, C. (2009). Detection of malicious code by applying machine learning classifiers on static features: A state-of-the-art survey. *Information Security Technical Report*, *14*(1), 16-29.

Shin, W., Kiyomoto, S., Fukushima, K., & Tanaka, T. (2010, August). A formal model to analyze the permission authorization and enforcement in the android framework. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on* (pp. 944-951). IEEE.

Tzermias, Z., Sykiotakis, G., Polychronakis, M., & Markatos, E. P. (2011, April). Combining static and dynamic analysis for the detection of malicious documents. In *Proceedings of the Fourth European Workshop on System Security* (p. 4). ACM.

Van Randwyk, J., Chiang, K., Lloyd, L., & Vanderveen, K. (2008, October). Farm: An automated malware analysis environment. In *Security Technology, 2008. ICCST 2008. 42nd Annual IEEE International Carnahan Conference on* (pp. 321-325). IEEE.

Vasudevan, A. (2008, December). MalTRAK: Tracking and Eliminating Unknown Malware. In *ACSAC* (pp. 311-321).

Vidas, T., Votipka, D., & Christin, N. (2011, August). All Your Droid Are Belong to Us: A Survey of Current Android Attacks. In *WOOT* (pp. 81-90).

Vinod, P., Jaipur, R., Laxmi, V., & Gaur, M. (2009, June). Survey on malware detection methods. In *Proceedings of the 3rd Hackers' Workshop on Computer and Internet Security (IITKHACK'09)* (pp. 74-79).

© COPYRIGHT UPM

Wu, L., Ping, R., Ke, L., & Hai-xin, D. (2011, September). Behavior-Based Malware Analysis and Detection. In *Complexity and Data Mining (IWCDM), 2011 First International Workshop on* (pp. 39-42). IEEE.

Xie, L., Zhang, X., Seifert, J. P., & Zhu, S. (2010, March). pBMDS: a behavior-based malware detection system for cellphone devices. In *Proceedings of the third ACM conference on Wireless network security* (pp. 37-48). ACM.

Zhao, M., Zhang, T., Ge, F., & Yuan, Z. (2012). RobotDroid: A Lightweight Malware Detection Framework On Smartphones. *Journal of Networks*, *7*(4), 715-722.

Zhou, Y., & Jiang, X. (2012, May). Dissecting android malware: Characterization and evolution. In *Security and Privacy (SP), 2012 IEEE Symposium on* (pp. 95-109). IEEE.

Zurutuza, U., Uribeetxeberria, R., & Zamboni, D. (2008, October). A data mining approach for analysis of worm activity through automatic signature generation. In *Proceedings of the 1st ACM workshop on Workshop on AISec* (pp. 61-70). ACM.

© COPYRIGHT UPM