Malicious Traffic Classifier in android using Neural Networks

Qamar Abbas1,* , Qamber Abbas2 and Jesse S. Jin2
1School of Computer Software, Tianjin University, Tianjin, China
2School of Computer Software, Tianjin University, Tianjin, China

*Corresponding author e-mail: qamarabbas@tju.edu.cn

Abstract. Now a day’s android is the fastest growing package in hand-held operating system. And it has become the most appealing and practical goal of malicious applications. This principal platform has approved itself not only in the mobile global but also in the Internet of the Things (IOT) devices. The most real risk of Android clients is malware contamination by means of Android application markets. The huge diffusion of malware in cellular platform is plaguing customers. Threat of malicious software has come to be an essential element within the protection of smartphones. The war between protection analyst and malware intellectual is abiding as contraption grows. The proposed methodology demonstrated the common precision 94% a tagged mobile malware datasets visitors with a lot of applications contains benign and twelve very different groups of all malware and adware in particular and complicated nature of malware is converting quickly and therefore emerge as harder to understand. We assessed different blends of oddity location calculations, include determination strategy and the quantity of the top highlights to decide the mix that gives the best activity in distinguishing new malware on Android. This research condenses the malware progression recognition procedures backed machine learning algorithms devoted to the Android OS.

1. Introduction

Smartphones emerged as a state of art facility to serve the purpose. Their powerful sensing, socializing and networking abilities make them unbeatable and stronghold of user’s attraction. Additionally, many surveys have validated their popularity repeatedly. Smartphones have practiced intense extension, embracing the corporate intranets as well as Internet. Devised as programmable, vulnerable, networked strategies, smartphones are persuaded to different malware dangers and are prone to several malware risks such as Trojan horses, viruses, and worms, each of them are recognized though desktop policies. These devices permit users to approach and look through the Internet, accept and send emails, MMSs, and SMSs, get along with other devices for exchanging information, and trigger numerous applications, that make these devices hit targets. One of the cell phone’s key features is that a device can run a huge kind of unit software on an application runtime surroundings. However, when a malware is setup at the telephone, the telephone consumer sides a critical danger such as information escape, reaching root concession and abuse purposes of the smartphone as with the state of dealings of malware infection mobile. Malware are malevolent software used to gather data and/or assistance entry to mobile computer devices containing smartphones or tablets. Especially, they are presented and reorganized with third- party bids to insert malicious subject right into a phone and therefore disclose the device’s defence. A negotiated smartphone can cause intense compensations to both the
cell shipper and users. Malware in smartphone can render it partially or completely useless; steal personal figures (most likely through Social Engineering); purpose discarded billing; or pervert each term in a customer’s phonebook.

Viable assault paths into smartphones that comprise though: internet connections (via 3G or Wi-Fi network access); cellular networks; USB/ActiveSync and different peripherals. The challenging circumstances for telephone defence have become very fair. The framework depends on a trivial application, installed on the mobile device that models numerous system metrics and investigates them in order to make implications about the well-being condition of the device. Moving towards the detection and prevention techniques. In literature, previously analysis is done in the form of static, dynamic or a combination of both static and dynamic to detect and analyse Android malware.

2. Related Work

2.1. Malware Detection Techniques
Current PCs and correspondence edges are profoundly arranged to a few sorts of assault. A corporate method for inciting these assaults is by methods for malignant programming (malware, for example, worms, infections, and Trojan ponies, which, when spread, can source serious mischief to held clients, business organizations and organizations. In Dynamic Analysis (otherwise called social based examination) the presentation contains data that is gathered from the working framework during the execution of the program, for example, framework calls, arrange access and documents and memory changes.

In Static Analysis, data about the program or its anticipated direct contains unequivocal and understood translations in its twofold code.

2.2. Malware Detection in Mobile Devices
In most cases that examinations have offered and surveyed Host-based Intrusion Detection Systems (HIDS). In excess of a couple of frameworks are investigate in this segment. In 2012 Kuhnel and Meyer proposed a malware detection technique by usage of a sensing application. This technique includes 30 families of malwares belonging to Android platform. These families were divided into four types RAT (Remote Access Tool), HTTP based, SMS based, and Calls based. Also, a filtering component was added which works on user space of Architecture of Android which was used for network analysis and was controllable by the sensor app. The achieved an accuracy of 95% this research claimed that by filtering outgoing and ingoing traffic they can detect malicious activities. The research purposed a novel model which was based on feature set like bytes of data send or received, state of network, mode of send or receive, total amount of time and last active or modification time in
minutes also with aggregation functions like average, maximum, minimum and standard deviation, for representation of certain traffic features of each App.

3. DETECTION METHODS

3.1. Malware Detection Techniques

The context suggested employs a machine getting to know the procedures for recognizing a malware detection device. In such an attack, the malware detector ad infinitum monitors numerous capabilities and complaints obtained from the scheme and then smears regular device getting to know classifiers to categorize amassed annotations as any normal (benign) or peculiar (malicious). Set up on earlier functionality and later comparing the aid consumption, difficulty, we found to evaluate the subsequent applicant classifiers: okay-why, histograms, selection tree, logistic regression, naïve Bayes and Bayesian networks. Accordingly, it's miles crucial to acknowledge the actual path of the function directions inside the test-set with a view to companion the actual (I. E., actual) magnificence with the magnificence created through the proficient classifier as show in Fig. 2 sequence diagram explaining the sequence of the main situation of the malware classifier.

Figure 2. Sequence diagram describes the sequences of the main scenario of the malware classifier.

3.2. Feature Selection

In machine Learning solicitations, a massive variety of mined features, a number of which unrelated or redundant, numerous current troubles including, unreliable learning algorithm, over-fitting, dropping generalization, and growing version difficulty and run-time. The Dataset provided by Canadian Institute of Cyber Security contains ppcaps of Bengin, Adware and Malware traffic. Weka was used to perform some data cleansing on these combined CSV’s to remove instances having NULL values. The output dataset with all three categories contains more 600k instances each having 76 features.
4. Evaluation

The good way to gauge our malware detection framework, we accomplished greater than few experiments. For you to perform the evaluation between the numerous detection algorithms and function choice schemes, we employed the following general metrics: True positive rate (TPR), that's the ratio of superb illustrations (i.e. feature vectors of malicious programs) classified as it should be. False positive rate (FPR), which is the ratio of bad illustrations (i.e., feature vectors of benign programs) misclassified and total accuracy,

\[
TPR = \frac{TP}{TP + FN} \quad (1)
\]
\[
FPR = \frac{FP}{FP + TN} \quad (2)
\]
\[
\text{Total Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)
\]

4.1. Creating the dataset for the experiments (Android Adware and General Malware Dataset)detection

The state-of-the-art and innovative android malware is capable of apprehend the prevalence of the emulator utilized by the malware professional and in reply, alter its conduct to avoid detection. AAGM dataset is captured by using connecting the android apps on the actual smartphones semi-automated. The dataset is constituted of 1900 packages with the subsequent 3 classes:

1. General Malware, which accommodates of the subsequent:
   - AVpass: considered to be dispersed in the arrival of a chronometer app.
   - FakeAV: composed as a trick that setups customer to buy a complete shape of the product with a particular quit goal to re-intercede non-existing contaminations.
   - GGTracker: supposed for sms misrepresentation (sends sms messages to an exceptional charge range) and facts taking.

2. Adware
   - Airpush: designed to carry spontaneous notices to the consumer's frameworks for data taking.
   - Dowgin: designed as a observe library which could likewise take the purchaser's facts.
   - Mobidash: designed to expose advertisements and to trade off patrons near domestic statistics.

3. Benign
   The dataset include the following:
   - Pcap files – the network site visitors of each the malware and benign (20% malware and 80% benign).
   - Csv files - the listing of extracted network visitors’ functions generated via the CIC-flowmeter.

5. Conclusion

In this research, a machine learning based Android malware detection model is purposed this model works on 76 features for effective classification of Android malwares. Moreover, this framework uses TCP and UDP flows-based features which are divided into behavior based, time based, packet based, and flow based. The experimental analysis of purposed model depicts that purposed model have a high accuracy of 94% having false positive rate of 0.08%. Also, passive testing of model on live traffic shows that, model is performing pretty well on live scenario also. The purposed model is also speed efficient and exploit parallelism which is good for handling high bandwidth of network traffic. The Algorithm used also 10x less time in training and testing as compared to another Machine learning Algorithm.
6. Acknowledgments

This work was partially funded by college of software, Tianjin University and through by scholarship of Chinese government council (CSC) and by support of government of Pakistan scholarship. We acknowledge the support of different scholarships councils.

References

[1] Shabtai, A., Moskovitch, R., Elovici, Y., & Glezer, C. (2009c). Detection of malicious code by applying machine learning classifiers on static features: A state-of-the-art survey. Information Security Technical Report, 14(1):1–34.

[2] Yin, H., Song, D., Egele, M., Krugel, C., & Kirda, E. (2007). Panorama: Capturing system-wide information flow for malware detection and analysis. In ACM conference on computer and communications security.

[3] Shih, D. H., Lin, B., Chiang, H. S., & Shih, M. H. (2008). Security aspects of mobile phone virus: A critical survey. Industrial Management & Data Systems, 108(4), 478–494.

[4] Shabtai, A., Fledel, Y., & Elovici, Y. (2009a). Detecting malicious applications on android by applying machine learning classifiers to static features (Poster). Presented in the 25th annual computer security applications conference (ACSAC). Honolulu, Hawaii.

[5] Shabtai, A., Fledel, Y., Kanonov, U., Elovici, Y., Dolev, S., & Glezer, C. (2010b) Google android: A comprehensive security assessment. IEEE Security and Privacy Magazine. doi:10.1109/MSP.2010.2.

[6] Muthukumaran, D., et al. (2008). Measuring integrity on mobile phone systems. In Proceedings of the 13th ACM symposium on access control models and technologies.

[7] Moreau, Y., Preneel, B., Burge, P., Shawe-Taylor, J., Stoermann, C., & Cooke, C. (1997). Novel techniques for fraud detection in mobile telecommunication networks. In ACTS mobile summit.

[8] http://www.unb.ca/cic/datasets/android-adware.html

[9] G.J. Tesauro, J.O. Kephart, and G.B. Sorkin. Neural networks for computer virus recognition. IEEE Expert, 11(4):5 –6, aug 1996.

[10] Yajin Zhou, Zhi Wang, Wu Zhou, and Xuxian Jiang. Hey, you, get off of my market: Detecting malicious apps in official and alternative android markets.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Duchesnay E. Scikit-learn: Machine Learning in Python . Journal of Machine Learning Research, 12:2825–2830, 2011.

[12] Ali-Gombe, A., Ahmed, Irfan, Richard III, Golden G., Roussev, Vassil, 2015. Opseq: android malware fingerprinting. In: Proceedings of the 5th Program Protection and Reverse Engineering Workshop, PPREW@ACSAC, Los Angeles, CA, USA, December 8, 2015, pp. 7:1e7:12. https://doi.org/10.1145/2843859.2843860. http://dblp.org/rec/bib/conf/acsac/GombeARR15

[13] Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Gregory S., Dean, Jeffrey, 2013. Distributed Representations of Words and Phrases and their Compositionality. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States, pp. 3111e3119.

[14] Karbab, E. B., Debbabi, M., Mouheb, D., Fingerprinting Android packaging: Generating DNAs for malware detection, Digital Investigation. Kim, Y., Convolutional neural networks for sentence classification, CoRR

[15] Tensorflow, 2017. https://www.tensorflow.org. Vidas, T., Tan, J., Nahata, J., et al., 2014. A5: automated analysis of adversarial android applications. In: Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices. Wei, T.-E.E., Mao, C.-H.H., Jeng, A.B., Lee, H.-M.M., Wang, H.-T.T., Wu, D.-J.J., 2012. In: Trust. Secur. Priv. Comput. Commun. Yan, L., Zhang, Y., et al., 2008. The Internet of Things: from RFID to the Nextgeneration Pervasive Networked Systems. Auerbach Publications. Yuan, Z., Lu, Y., Wang,
Z., Xue, Y., 2014. Droid-Sec: deep learning in android malware detection. In: ACM SIGCOMM Comput. Commun. Rev

[16] Android. https://www.android.com/. Security and Permissions. http://developer.android.com/guide/topics/security/security.html.

[17] Android.Geinimi, http://www.symantec.com/security_response/writeup.jsp?docid=2011-010111-5403-99.