# Security Evaluation Metrics implemented through Expert System for Android Devices

Divesh Badod
Department of Computer Science,
Rochester Institute of Technology,
Rochester, NY, USA
db9932@rit.edu

***Abstract*—Growth of smartphone usage in recent years has led to a tremendous surge in the collection of personal and public data via social media applications, gaming applications and various other applications requiring you to fill in such data to learn what type of user you are and make your life easier with subsequent follow-up services. All of this traffic in data has led to the evolution of Big Data. This inevitably leads to the question of security of the said data whether it is in your smartphone or your personal computers this data is stored in your hardware without your knowledge and any sense of secure wall protecting it. This paper purposes a security system built around the idea of evaluating the security of the data, with the proposed metrics. The Expert system being implemented will be incredibly helpful in establishing the rules for ambiguous situations. After establishing the rules we will be ensuing with a machine learning technique using Artificial Neural Network which will evaluate the output generated from the Expert-System and generalize the predictions of said rules for all the android devices.**

***Keywords*— *Big-Data, security evaluation, Expert-Systems, Machine Learning, Artificial Neural Networks, Metrics and Android Devices.***

## I. Introduction

The idea of having a handheld computer has led to the boom in smartphone industry resulting in 3.8 billion smartphones in use as of now[4]. By just considering the number of smartphones being used today the amount of data generated and stored in said smartphones is insurmountable. Naturally, the question of security will arise as these data are private and can be leveraged for unscrupulous deeds. Android devices are dominating the market today with nearly 88% of users using this OS or owning an android phone and other 12% being iOS users[6].

This paper will introduce you to the metrics which I have siphoned out covering the overall general security questions of threats which can be induced on mobile devices. Prime targets can be Data, Identity and Availability, and the threats that attack these targets can be viruses, DOS, Spyware, Malware, Botnets, Trojans, Ransomware, Phishing to name a few[5]. Every threat originates via an online source or unknown applications or even antivirus software. These vulnerable sources can be identified before they become vulnerable and the metrics proposed in these papers help us in identifying them. These metrics will generalize the usage of this Expert-System on different android devices or even computers because every android device or computer will indisputably have the sources mentioned above.

Expert-Systems, as the name suggests, are experts in knowledge provided in the form of rules in if-then context. These systems are emulators of human experts who help in solving complex problems through their decision-making abilities. Metrics in our Expert-System will play the role of facts which will be then used to apply a set of rules which in turn will be used for a final evaluation of the device security presenting on a scale of 1-10. Later on, the data collected after implementing the expert system will be used to train and test the neural network which can make the product more reliable as it won't be dependent on only the metrics but on itself, hence making the application much more compatible with other devices and not only android.

## II. Literature Review

The paper [2] infers that due to this rise in the global market for smartphones there has also been a rise in low-end smartphones with manufacturers being paid to pre-install applications with malware and sell it, so that the attackers have an incentive to attack the devices. Such practices have led to a heave in distribution channels of devices with malware pre-installed in it, and such channels being phased out of the picture because of such high demands in the market for cheaper smartphones makes it difficult to track down the source of malware attackers. The aforementioned paper hence introduces a security evaluation system for android devices with a focus on android firmware by detecting malware considering 3 factors, System Signature Vulnerability detection, Network Security Vulnerability detection, Privilege Escalation Vulnerability detection. Briefly scrutinizing these vulnerabilities will reveal that they cover all the areas susceptible to malware attacks, hence making a pre-emptive strike in such areas via this evaluation system can considerably help in securing the android firmware and therefore the phone.

The paper by Reijo Savola from VTT Technical research centre in Florida[1] describes the process of carefully selecting security metrics before making an evaluation system. Constituting that security evaluation is an iterative process based on security requirements, metrics, and evidence collection, the gist of this paper covers how important it is to elaborate the roles of security metrics before deciding which to choose. Security evidence is a vital constituent for designing a security evaluation system and even though the process of collecting such evidence is underdeveloped as mentioned in the paper, 3 components help us simplify the answer to the questions regarding the vulnerabilities of the said system namely, evidence collected during research and development, security evidence during implementation pointing out the vulnerabilities during the implementation phase and maintenance evidence pointing out the security threats during maintenance. After gathering all such evidence the evaluation system development begins its iterative process again, analysing all the threats, prioritizing requirements then modelling the behaviour of our system and gathering evidence. Metrics, on the other hand, play a role in measuring the activity performed. Security metrics and measurements provide us with the information which helps us in the decision-making process especially in cases of assessment and prediction. These metrics can be investigated in the following classifications, Quantitative vs Qualitative, Objectivity vs Subjectivity, Direct and Indirect, Static vs Dynamic, Absolute vs Relative. These classifications provide us with the confidence of choosing the appropriate metrics depending upon the system and the evidence collected thereby facilitating the compatibility of the evaluation system.

Android Enterprise Security White Paper[8] is the official report of android's security policies and Google's attempt to make users' smartphones as secure as possible. Covering every aspect of a smartphone's vulnerabilities such as device integrity, data protection, network security and security patches or updates, this paper has given a comprehensive idea of deciding how and which metrics are to be selected for the proposed security system along with the explications about the inbuilt security systems like Google Play Protect and Safety Net.

## III. Metrics

By analysing and studying the aforementioned papers the following metrics were decreed for an overall consideration of the security evaluation system. The metrics are as followed:-

TABLE I: Metrics

| Metrics | Contributions | Weights |
|---|---|---|
| Current Software in use | Checking if the latest update of the software is installed | 0.3 |
| Password Protected | Checking if the device is password or fingerprint protected | 0.5 |
| Firewall | Checking if the latest firewall is installed | 0.4 |
| Number of users | Checking how many users have access to this machine | 0.1 |
| Security Patches | Checking if latest security patch is installed | 0.3 |
| 3rd Party Apps | Checking how many 3rd party applications have been installed | 0.4 |
| System Architecture | Checking which system architecture is the device running on | 0.2 |
| Antivirus | Checking if the latest antivirus software has been installed | 0.3 |
| Browsers | Checking which browsers are safe to use | 0.2 |
| Peer to Peer Network | Checking if you use Peer to Peer networks | 0.3 |

## IV. EXPERT SYSTEM AND HIERARCHY OF RULES (PHASE I)

Expert systems in a broader sense emulate human experts in decision making given the knowledge represented in if and then rules[7]. Expert Systems are divided into subsystems called Inference Engine and Knowledge Base. The Inference Engine is the component that applies logical rules to the knowledge present in the expert system. This knowledge present in the Expert System is called Knowledge Base. Inference engines have two modes of applying rules first one called Forward-Chaining and second one called Backwards-Chaining. For our purpose of a security evaluation, we will be using forward chaining process.

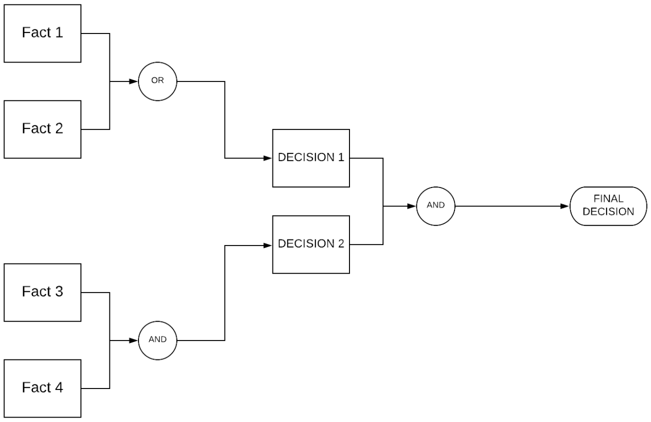

Fig. 1: Forward Chaining

In our case, the rules are the contributions of the metrics. For example, if the latest system is not installed in your device then the rule tells that its contribution to securing the device is reduced to 60% and knowledge in this example is that this device doesn't have the latest system installed. Through forward-chaining, the expert system will arrive at a point at which the weighted average of the metrics will be calculated and the security of the device will be evaluated based on those calculations. In order to implement the forward-chaining process, we have to set a hierarchy for the metrics in order to distribute the metrics in a format that covers the overall vulnerabilities of a device (Fig 2.).

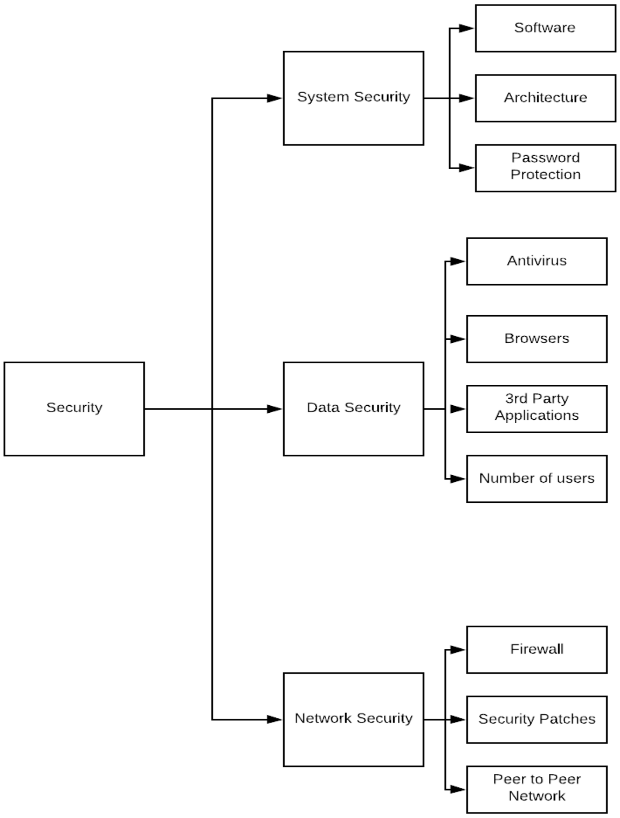

Fig. 2: Metrics Hierarchy Rule based

There is an assortment of expert systems at hand today and for this project, I'll be implementing the JESS expert system. JESS has extreme portability since it is written in JAVA, features also include Lisp-like language, JAVA API and graphical shell. And since android is more JAVA oriented JESS is the most reliable engine to choose. Along with JESS, I'll be using JAVA frameworks like Swing and AWT for GUI development of the system, where some of these metrics will be filled out manually and some will be collected automatically. Thus the phase-I of the project concludes with deciding and structuring the metrics for optimal solutions, implementing an expert system using given metrics and developing rules accordingly and a rudimentary GUI for better and uncomplicated interaction with the security system. 8

## V. Machine Learning (PHASE II)

The data collected from phase I will be the essence of machine learning implementation in phase II. Machine learning is described as the study of algorithms which automatically improves through experience[9]. There are 3 machine learning techniques Supervised Learning, Unsupervised Learning and Reinforcement Learning. Since Supervised Learning includes data with both inputs and outputs and the model is trained using this data to make predictions within the confines of the given data, I'll be using this technique for the phase II part of the project. The mathematical model will be an Artificial Neural Network(ANN) to train and test the data and predict the security rating for different types of devices. Limitations of rule-based systems like slower processing and less efficiency in terms of self-improvement have encouraged me to use this model to cover the wider proportions of diverse android devices. The input layers and the hidden layers will be evaluated in an iterative process until the predictions are as accurate as the evaluations done by the expert system. Python will be used to develop the said neural network since this language is at the forefront in building mathematical models for machine learning, because of the inbuilt libraries such as KERAS and TensorFlow. The resulting data generated during this process of training and testing the neural network such as True positives, True negatives, False Positives, False Negatives Accuracy, Precision, Recall, percentage of data used as training, testing and maybe validation, all will be represented using matplotlib library in Python.

## VI. Conclusion

Throughout this paper, I have explained the process of building a security evaluation system in two phases. Phase I being scouting out reliable metrics, evaluating the importance of said metrics for security evaluation, preparing rules via an expert system using said metrics, using the hierarchy of the metrics to set the rules to generalize the evaluation for all android devices, using a GUI to make it easier to interact with the system, get the knowledge for expert system manually and automatically and collect data generated after implementing the expert systems.

Phase II will include building an artificial neural network, training this neural network on the collected data, testing the network to make it more accurate until the accuracy matches the output of the expert system from Phase I.

In conclusion, the system being deployed will give an extensive sense of the importance of security in the users' devices. Since the number of smartphones has risen considerably and today's fast-paced technologically forwarding world requires you to use smartphones on daily basis the need of securing the data gets suppressed in one's mentality but with a few steps taken in the right direction and considering how to keep security a number one priority and yet not to let it hinder with the user's day to day tasks, is what I kept in mind while designing this evaluation system.

## References

[1] Riejo Savola, *Information Security Evaluation based on Requirements, Metrics and Evidence Information*, VTT Technical Research Centre of Finland, Oulu, Finland.

[2] Min Zheng, Mingshen Sun, John C.S. Lui, *DroidRay: A Security Evaluation System for Customized Android Firmwares*, Computer Science & Engineering Department, The Chinese University of Hong Kong.

[3] Peter g. Neumann, *Computer system security evaluation*, SRI International Menlo Park, California.

[4] Statistics of smartphone users worldwide, *https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/*

[5] Mobile Security, *https://en.wikipedia.org/wiki/Mobile_security*

[6] Market share of Smartphone OS statistics, *https://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/*

[7] Ernest Friedman-Hill, *The Rule Engine for the Java$^{TM}$ Platform Version 7.1p2*, Sandia National Laboratories Jess®.

[8] Android Enterprise Security White Paper(2018), *https://source.android.com/security/reports/Google_Android _Enterprise_Security_Whitepaper_2018.pdf*

[9] Machine Learning, *https://en.wikipedia.org/wiki/Machine_learning*