International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Virus Detection System for Android Using ANN

Harshada S. Palve[1], Vrunda K. Bhusari[2]

[1]P.G. Student, Savitribai Phule Pune University, Department of Computer Engineering, BSIOTR, Wagholi, Pune, Maharashtra, India

[2]Assistant Professor, Savitribai Phule Pune University, Department of Computer Engineering, BSIOTR, Wagholi, Pune, Maharashtra, India

***Abstract:** Popularity of Handheld Device is changed nowdays. People uses smart phone for many purpose same as desktop computers like web browsing, social networking, online Banking etc. In smart phone market currently Android is the platform with highest share. Because of such popularity and open source nature of Android OS it is main source of attacker's. As Android itself a very secure operating system so main source of viruses in android is the applications which is downloaded by user from internet. Here we proposed a system which is capable of detecting that whether application is malicious or not. And set of permissions is a dataset. Artificial Neural Network (ANN) algorithm is used as classifier. Set of permission is given as input to train the network. Output produced is the either "1"or "0".For analysis purpose to show that how viruses are propagated two applications are developed which uses Wi-fi and SMS facility of mobile phone.*

Keywords: Android, mobile, classifier, viruses

1. Introduction

The number of Android device users 2012 is 181 million. This represents approximately 75% of Smartphone users. As the number of Android device users increases, so too does the number of available applications. There are currently over 600,000 applications in the Play Store, which is used by Android user to download the android applications. Currently many handheld devices use android operating system because its openness and accessability. Android use application permissions or review processes to prevent the spread of malware while installing application.

The android smart phones are mostly targeted by the attackers, because open platform is provided by android developers [2]. Viruses writers do not want their malicious creation to be detected, analyzed as well as filter when they spread across network. In network topology there is two channels responsible for propagation of viruses i.e. Wi-Fi channel and SMS channel. Viruses based on Wi-Fi spread in mobile phones which are within communication range of infected phone.SMS based viruses steal the user personal information and send this data to the phones which are in address book of phone.SMS based viruses are more dangerous than other type of viruses. Human behavior also effects the dynamic propagation of viruses. Such as Operational behavior is primary factor contributing to SMS-based virus propagation [5].

Android consist of different layers such as Application layer, application framework layer, android Runtime layer and Libraries layer. Each layer comes with its own security mechanism.

1) Application layer (Android Permissions): Every application comes with a file named AndroidManifest.xml file contains the permissions that the application may require during execution. While installation the user is asked to grant all the permissions specified in the manifest.
2) Application Framework(Permission Enforcement):Services at this layer enforces the permissions specified in the manifest and granted by the user during installation.
3) Runtime (VM Isolation): Every android application is executed in a separate Dalvik Virtual Machine. This provides isolation among applications.

There are three types of threats as follows:
1) Malware: Malware gains access to a user device for the stealing of private data, damaging the device, reducing battery lifetime. The attacker takes the advantage of device vulnerability. This threat includes Viruses, botnets,Warms and Trojans
2) Personal Spyware: Spyware collects personal information such as location or text message history. With personal spyware, the attacker has physical access to the device and installs the software without the user's knowledge. Personal spyware sends the information of victim to the person who installed the application onto the victim's device, rather than to the author of the application.
3) Grayware: Some legitimate applications collect user data for the purpose of marketing or user profiling. Application markets may choose to remove or allow grayware when detected on a case-by-case basis.

2. Literature Review

Smartphone have recently become very popular because they provide many facilities by integrating traditional mobile phones with handheld computing devices. However the exibility of running third-party software also leaves the smartphone open to malicious viruses. In [1] author proposed SmartSiren, which is virus detection and alert system for smartphones. For detecting the viruses SmartSiren collects the communication activity information from the smartphones, and further performs joint analysis to detect both single-device and system-wide abnormal behaviors of smart phone. Proxy-based architecture used to offload the processing burden from resource-constrained Smartphone and simplify the collaboration among smartphones. When any potential virus is detected, the proxy quarantines the out-

break by sending targeted alerts to those immediately threatened smartphones.

Cell phones are increasingly targeted by various worms, which cause the leakage of user's private information, extra charges, unauthorized access to device phonebook, and depletion of battery power etc. Work which is presented In [2], observed the propagation dynamics of cell-phone worms, which exploit Multimedia Messaging Service (MMS) and/or Bluetooth for spreading. And then systematic countermeasure against the worms are proposed At the terminal level, Graphic Turing test is adopted and identity-based signature is used to block unauthorized messages leaving from compromised phones; at the network level, push-based automated patching scheme for cleansing compromised phones is used. Through experiments on phone devices and a wide variety of networks, it is show that cellular systems taking advantage of proposed defense can achieve a low infection rate (e.g. less than 3% within 30 hours) even under severe attacks. In this article systematic solution is proposed which include both terminal-level and network-level preservation these solutions, however, are still not complete because they do not leverage collaborations between the terminals and the network to throttle worm spreads in a systematic way.

Many emerging malware can utilize the proximity of devices to propagate in a distributed manner; Community-based Proximity Malware Coping scheme i.e.CPMC.It utilizes the structure of Social community, which gives a stable and controllable aspect of security, in smartphone-based mobile networks. In [3] a closeness-oriented delegation forwarding scheme combined with a community level quarantine method is proposed as the short-term coping components. These components contain a proximity malware by quickly propagating the signature of a detected malware into all communities while avoiding unnecessary redundancy. Figure1 shows that Proximity malware Propagation. Malware Can Propagate through Bluetooth when two nodes are in geographic proximity.

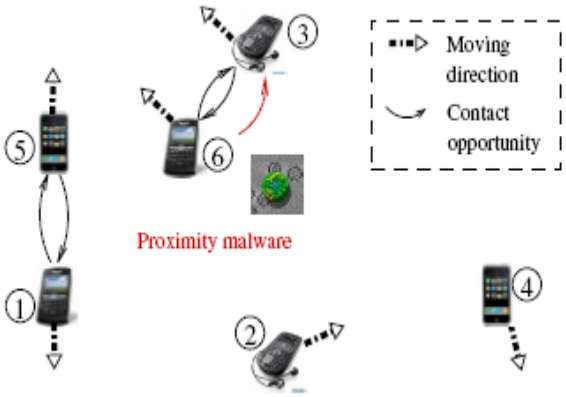


Figure 1: Proximity malware propagation [3].

In [4] author represents some techniques which are helpful for finding mobile malware targeting depletion of battery energy. Such type of malware and viruses are generally difficult to detect and prevent. Framework which is proposed by author in this paper consist of two main component i.e. power monitor and data analyzer. Power monitor collect power samples and build power consumption history from that samples, and data analyzer generate a power signature from constructed history.

3. System Implementation

3.1 System Architecture

The Figure 2 shows the overall system architecture. System proposed here is deployed on android mobile devices. As per shown in figure the process flow goes according to the system architecture. The system architecture includes tasks which are: i) Evaluate, ii) Permissions, iii) Data Sets, iv) Classifier, v) Model. Android Phone includes thousands of applications. All applications run in sandbox which is isolated area of system that does not having access to system Resources. If user gives the permission to install the application then that particular application able to use system resources. Main tasks of proposed system are explained as follows:

i) **Evaluate**: When user wants to download any application then he request for link of the applications. All application links is stored at server side where log of all application is maintained. So all applications are evaluates at server side.

ii) **Permissions**: Every application having its associated permissions. Permissions are used for application to give access to particular system resources like game application may need to enable vibration and sound but does not require to access user phone book or Internet. When user request for particular application then server send the appropriate link along with associated permissions. If user agrees about all permissions then application is being installed. Attacker cannot create new code he just modify existing code. And while installing applications send illegal set of permissions.

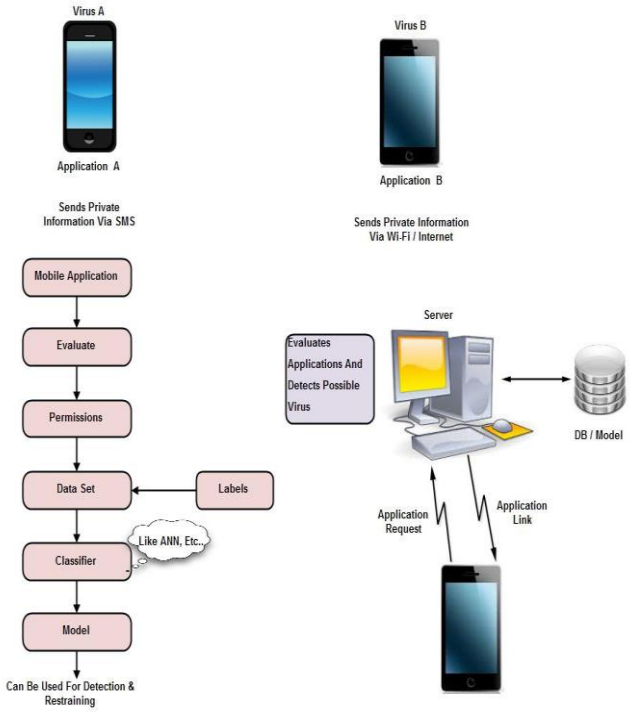


Figure 2: Detail Architecture of System Design

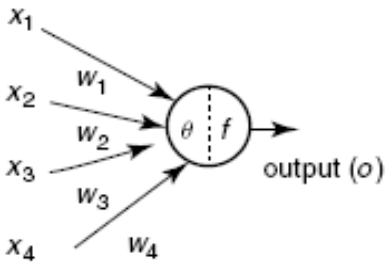
iii**) Data Set**: Data set consist of all applications and relevant permissions.

iv) Classifier: Artificial Neural Network is used as classifier. Input provided is Set of permissions and output is produced as "0" or "1" which indicates that whether application is malicious or not. Neural network is trained according to that it produced output.

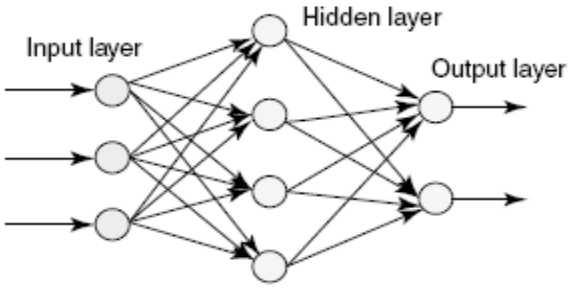
3.2 Algorithm

Algorithm used is Artificial Neural Network (ANN) which is used as classifier. ANN is same like biological Neuron. Artificial neurons, is a basic processing elements of neural network. In a mathematical model of the neuron, a connection weight represents the effect of synapses that modulate the effect of the input signals; transfer function represents the nonlinear characteristics of neurons. The neuron impulse is then computed as the weighted sum of the input signals. The learning capability of an artificial neuron is achieved by adjusting the weights in accordance to the chosen learning algorithm.

Figure3 shows the A typical artificial neuron and the modeling of a modeling of a multilayered neural network. X1, X2, X3…….Xn is a input to the neuron and W1, W2, W3, W4…Wn is the weight associated with neuron. The neuron output signal *O* is given by the following relationship:



(a) Artificial Neuron



(b) Multilayered Artificial Neuron Network

Figure 3: An artificial neuron architecture and multilayered artificial neuron network.

$$O = f(net) = f\left(\sum_{j=1}^{n} w_j x_j\right)$$

Where Wj is the weight vector and the function f (net) is referred to as an activation function. The variable net is defined as a scalar product of the weight and input vectors.

$$net = w^{\mathrm{T}} x = w_1 x_1 + \cdots + w_n x_n$$

3.2.1 Back Propagation Algorithm

Actually, Back Propagation is the training or learning algorithm. Back Propagation network learns by example. Input given to the algorithm is examples of what we want the network to do and it changes the network's weights so, when training is finished; it will give the required output for a particular input. Suppose neural network is as shown in Figure4 here only connection between A and C is considered. This back propagation algorithm for Figure4 work as follows:

Step 1: First apply the inputs to the network and work out the output this initial output could be anything, as the initial weights were random numbers.

Step 2: Next work out the error for neuron B. The error is *what we want – what we actually get.*

$$Error_B = Output_B\{1 - Output_B\}\{Target_B - Output_B\}$$

Step 3: Change the weight. Let W_{AB}^{+} be the new (trained) weight and W_{AB} be the initial weight.

$$W_{AB}^{+} = W_{AB} + \{Error_B \times Output_A\}$$

Step 4**:** Errors for the hidden layer neurons is calculated Error at hidden layer cannot calculate directly unlike the output layer because we don't have Target, so we Back Propagate them from the output layer (hence the name of the algorithm). This is performed by taking the Errors from the output neurons and running them back through the weights to get the hidden layer errors. Such as if neuron A is connected to B and C then we take the errors from B and C to generate an error for A.

$$Error_A = Output_A\{1 - Output_A\}\{Error_B W_{AB} + Error_C W_{AB}\}$$

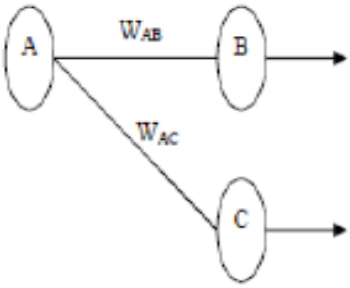


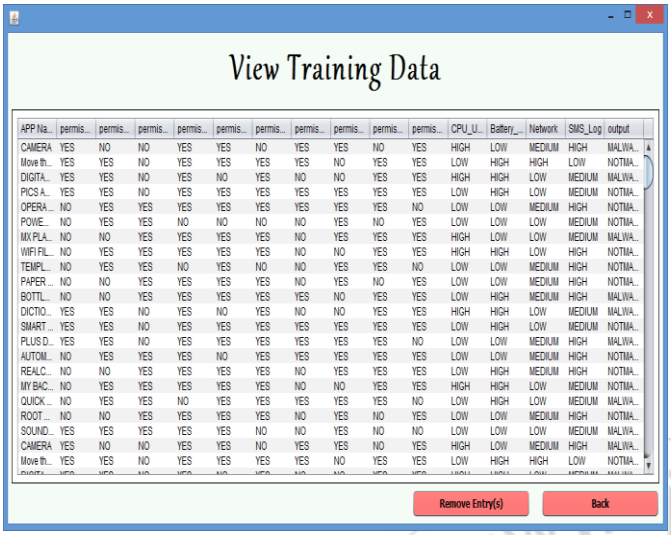
Figure 4: Single connection learning in back propagation network

4. Results

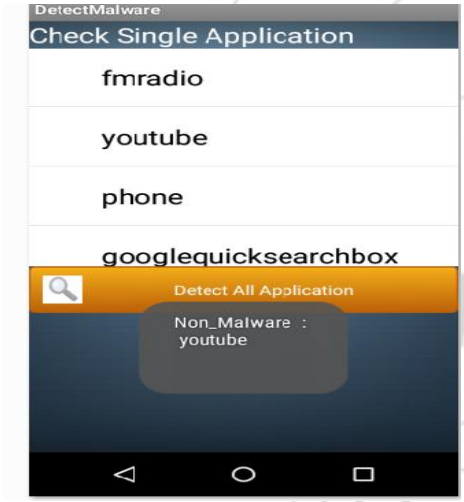
Platform used for system development is Windows XP/7, JDK Android SDK, Eclipse and Database used is Apache Tomcat where all applications and respected permissions are

stored. On generated results different types of testing are performed. Snapshots of Results are described below.

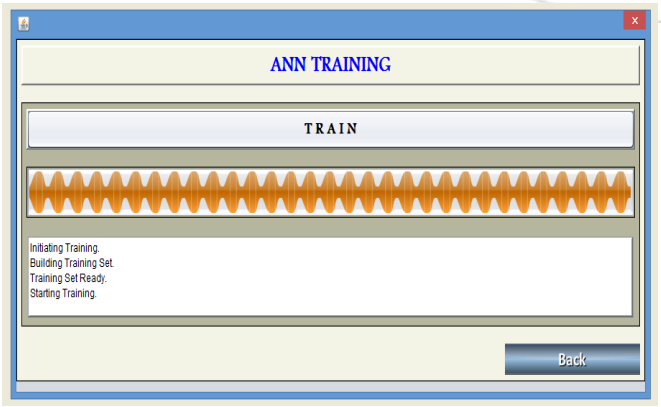
4.1 Data Set used for detections of malicious applications



4.2 Malicious Application Detected

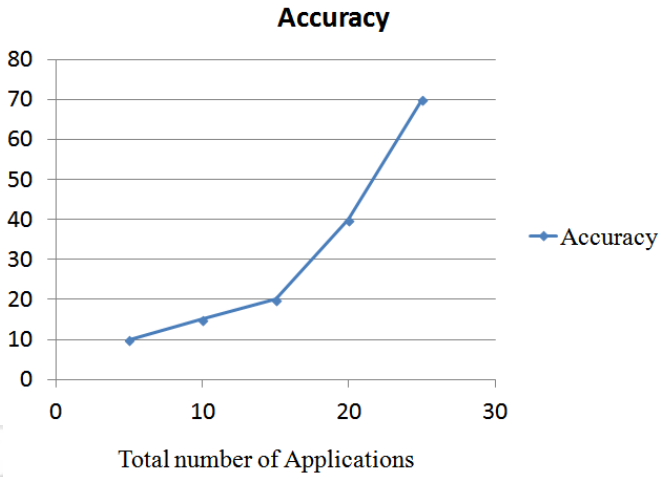


4.3 Artificial Neural Network training



4.4 Graph

As Number of applications stored in database are increases then more training provided to the ANN. So accuracy will be increased.



5. Conclusion

Viruses spread from one mobile network to another and causes the many damage in smart phones. In this situation it is necessary to understand the spreading pattern of viruses in order to control virus propagation. System which is proposed first stored user application log in database at server side. In this database classification is performed and possible viruses which are present in application are detected. Artificial Neural Network (ANN) algorithm is used as Classifier; learning is performed to train the artificial neural networks. Two layer network model used for simulating and analyzing the propagation dynamics of SMS based and Wi-Fi based viruses. Along with these two types of human behaviors are consider which is based on some empirical studies and statistical data. As for our future work, we will investigate the hybrid viruses that propagate through both Wi-Fi and SMS channels. In next step, we will extend our model to incorporate additional characteristics of human mobility and operations.

Acknowledgement

I would like to thank my sincere gratitude to my guide Prof. Vrunda K. Bhusari for her kind patience, support, motivation, and immense knowledge. Her guidance helped me in all the time of research and writing of this paper.

References

[1] J. Cheng, S.H.Y. Wong, H. Yang, and S. Lu,"Smartsiren Virus Detection and Alert for Smartphones." Int'l Conf. Mobile System,Applications,and Services (MobiSys'07), pp.258-271,2007

[2] L. Xie, H. Song, T. Jaeger, and S. Zhu, "A Systematic Approach for Cell –Phone Worm Containment,"Proc 17th Int'l World Wide Web Conf.(WWW'08),pp.1083-1084,2008.

[3] F. Li, Y. Yang, and J. Wu, "CPMC: An Efficient Proximity Malware Coping Scheme in Smartphone-Based Mobile Networks," Proc. IEEE INFOCOM, pp. 2811-2819, 2010.
[4] H. Kim, J. Smith, and K.G. Shin,"Detecting Energy-Greedy Anomalies and Mobile Malware Variants," Proc. Sixth Int'l Conf. Mobile Systems, Applications, and Services (MobiSys 08), pp. 239-252, 2008.
[5] S. Cheng, W.C. Ao, P. Chen, and K. Chen, "On Modeling Malware Propagation in Generalized Social Networks," IEEE Comm. Letters,vol.15, no.1, pp.25-27,Jan. 2011.
[6] Bose, X. Hu, K.G. Shin, and T. Park, "Behavioral Detection of Malware on Mobile Handsets," Proc. Sixth Int'l Conf. Mobile Systems, Applications, and Services (MobiSys '08), pp. 225-238, 2008.
[7] Chao Gao and Jiming Liu,"Modeling and Restraining mobile Virus Propagation", IEEE transactions on Mobile Computing, Vol 12, No.3

Author Profile

Ms. Harshada S. Palve received her Bachelor of Engineering(Information Technology) from MIT AOE Alandi, Pune and now she is pursuing her Master of Engineering(Computer Engineering) from BSIOTR Wagholi, Pune, Maharashtra..

Prof. Vrunda K. Bhusari received her M.Tech(Computer Engineering) from Bharati Vidyapeeth, Pune and now she is working as Assistant professor, Department Of Computer Engineering, BSIOTR Wagholi, Pune, Maharashtra.. Her research areas include Network Security.