IEEE Sponsored 2nd International Conference on Innovations in Information, Embedded and Communication systems (ICIIECS)2015

Cloud Based Intrusion Detection Architecture for Smartphones

Manish Kumar
Assistant Professor, Dept. of Master of Computer Applications, M. S. Ramaiah Institute of Technology, Bangalore
Research Scholar, Department of Computers Science and Applications,
Bangalore University, Bangalore, INDIA
E-Mail:- manishkumarjsr@yahoo.com

Dr. M. Hanumanthappa
Professor, Dept. of Computer Science and Applications,
Jnana Bharathi Campus, Bangalore University,
Bangalore -560 056, INDIA

***Abstract*- Smartphones are phones with advanced capabilities like those of personal computers (PCs). Smartphone technology is more and more becoming the predominant communication tool for people across the world. People use their smartphones to keep their contact data, to browse the internet, to exchange messages, to keep notes, carry their personal files and documents, etc. Users while browsing are also capable of shopping online, thus provoking a need to type their credit card numbers and security codes. As the smartphones are becoming widespread, it's also becoming a popular target for security threats and attack. Since smartphones use the same software architecture as in PCs, they are vulnerable to be exposed to similar threats such as in PCs. Recent news and articles indicate huge increase in malware and viruses for operating systems employed on smartphones (primarily Android and iOS). Major limitations of smartphone technology are its processing power and its scarce energy source since smartphones rely on battery usage. The smartphones have less storage and computational power to put into effect highly complex algorithms for intrusion detection and implementing signature based attack detection. Now in this paper, we propose a cloud based Intrusion Detection System for smartphones to overcome the issues of smartphone resource constraints and to detect any misbehavior or anomalous activity effectively.**

***Index Terms*—Intrusion Detection System, Signature Database, SNORT, Smartphone**

I. Introduction

As we place more and more information and rely on smartphones, they become soft targets for information and identify theft,as well as denial of service attacks (e.g., battery exhaustion). The major problem with the Intrusion Detection Systems whichis developed for mobiles are basically based on the general computer and network technologies. These IDS architecture are not best suited for smartphones as it's require more CPU and memory consumption. Since smartphones and other mobile devices have limited energy resources, IDS implementation for smartphones become a challenging issue. The other major problem is that such protection systems need to be continually update their signatures from the central repository and since updating of phone signatures is energy-expensive it is more likely that the attackers may try to use newer kinds of attacks to compromise smartphones.

Other problem is that since there are many kinds of machines and certain attacks can target specific kinds of operating systems and machines which might not be a priority for IDS vendors and security companies that serve updates to their customers. Often, smartphones fall into this category. Because of all these problems there is a need for more general approach to solve these issues.

The majority of work that deals with mobile IDS systems deals with host based IDS in which methods of anomaly or rule-based are utilized and used to extract and perform analysis of the features and then make decisions on the state of the device. The extraction is done locally and the analysis is done either locally or on a remote server. Any smartphone IDS that would require high amount of computation on the device would lower the user experience, consume power resources, so the IDS that would truly be functional and protective for the user is extremely challenging to design and would have to be done on as lowest level OS as possible, carefully programming for evasion of memory leaks, etc.

II. Need For Cyber Security of Smartphones

A large number of smartphone malware have attempted to exploit vulnerabilities of smartphones. The new generation of smartphone malware attacks has increased in sophistication and is designed to cause severe financial losses and disruption of critical software applications.

Recent Smartphone security study shows that the mobile operating systems that are being targeted by malware, Trojan and viruses has significantly increased. As per the report released by Kaspersky [4], the highest number of attack is attempted on Android, attracting a whopping 98.05% of known malware. The reasons for this are Android's popularity and its market position. The prevalence of third party app stores and the fact that Android has an open architecture, making it easy to use for both app developers and malware authors alike.

Apart from invading privacy and security of the smartphone users, such threats also manage to form botnets by

978-1-4799-6818-3/15/$31.00 © 2015 IEEE

which communication infrastructures can face large-scale coordinated attacks.

As data transmission are becoming affordable and available, usage of smartphones for online financial transactions, social networking, mobile learning and web browsing become more easy and accessible, which may also cause several security issues. User can get malware on their smartphone just as it can bypass security and get on PC. Most of the attacks and malware are picked up from downloading a vulnerable files or by some activities on social-networking sites or visiting a phishing website. You can also get such attacks on your phone through shared links on mobile social networking apps and other peer-to-peer sharing applications. As architecture of such devices are much similar to classic personal computers in terms of functionality as well as performance, common security threats like worms, Trojans and viruses are also affecting smart phones. To protect smartphones in the same way as desktop-PC same security algorithms are required to be used. But these algorithms are highly resource consuming and can be complex too, so they cannot be executed on such smartphone as they are constrained by power, computational and storage limitations [1].

III. Related Work

There are lot of researchers have been contributing ideas to improve security system to prevent data loss and Intrusion Detection in Smartphone like Vadimir B. Oliveira et al.[20] via HoneyspotLabsac, a virtual honeypot for android which emulate intrusion detection on services like telnet, http and SMS.

Some researchers has provided their own security model. The permission-based security model is one of the most important security models in Android devices. The user could grant or deny the installation and the application itself specifies which resources of the device need to be used. Analysis and enforcement of this permission-based model has been proposed by various researchers.

Markus Miettine [10] et al. analyzed how the malware performed malicious activities on smartphone. They suggested that some information should be monitored for anomaly detection, such as operating system events, resource usage, application-level events, and so on. Finally, they came up with a unified intrusion detection model.

Samfat and Molva [5] propose architecture for mobile networks - IDAMN and they use anomaly detection methods as well as rule-based methods. There are three levels of detection; Location based detection (user located at two different locations at the same time), traffic anomaly detection (extremely low or extremely high levels network traffic), and the detection of anomalous behavior of specific mobile phone user.

Schmidt [3] et al. proposed a solution based on monitoring events occurring on Linux-kernel level. They use kernel system calls, network activity events and file system logs to detect anomalies in the system.

Schmidt et al. utilize a Symbian monitoring client for the smartphone based on Symbian OS, which collects and forwards collected features to anomaly detection server RADS. This data is then processed in order to distinguish between normal and abnormal behavior.

Shabtai et al. [2] proposed Adnromaly — a framework for anomaly detection on Android smartphones. It is host-based and it continually monitors various features and events obtained from mobile device and then apply machine learning anomaly detectors to classify the collected data as normal-benign, or abnormal-malicious [6][7].

There are many other researchers actively working on to address the issue of Intrusion Detection in smartphone but the major hurdle is that all the proposed solutions are resource intensive which is one of the biggest constrained for smartphones.

IV. Security Threats

The most frequent types of threats and infection channels, as well as corresponding security functions suitable to protect Smartphone are as follows:

1) Threats: - Denial-of-Service attacks against smartphones are carried out by flooding with a large number of packets to the device to consume system resources or battery. Information theft occurs when hackers attack smartphones to obtain personal information which might be sensitive or confidential. Theft-of-service occurs when malware uses smartphone resources, for instance, to send expensive SMS messages. Spam categorizes attacks where mobile users are targeted involuntarily with advertising, messaging, and other similar information.

2) Infection Channels: -Bluetooth viruses may infect mobile devices. The most well-known virus of this kind is Cabir. The Short Message Service (SMS) or Multimedia messaging service (MMS) can be used by smartphone viruses to spread within networks.

3) Security Function: -Since the sensitive or valuable information is being increasingly stored in smartphones and/or transmitted over the network, this data should be encrypted to ensure that the confidentiality of the information is not compromised.

V. Research Framework

The primary goal of this research work is to design and implement a prototype for the Smartphone Intrusion Detection. The main objective of our proposed solution is that, it should not be much resource intensive and feasible for implementation. It will contribute to regulate data in mobile computing on Smartphone, especially android mobile system. The proposed solution will detect attacks (viruses, worms, Trojan horses and metamorphic malware) and prompt users to take actions to prevent breaches[8].

Furthermore, this project will also extend the existing knowledge about Android Smartphone's security and provide in-depth understanding of how to effectively manage emerging threats and attacks.

VI. Proposed Solution and Implementation

In our solution (Figure 1) we are proposing the cloud-based IDS for Smartphone. It consists of a cloud-based service which would allow users to install a light-weight agent on their Smartphone and register to an online cloud-service by specifying their operating system, applications installed on their phone and other relevant information about their device. Afterwards, this specific Smartphone is emulated in a virtual machine on the cloud using a proxy which duplicates the incoming traffic to the device and then forwards the traffic to the emulation platform, where detection is performed.

The system is developed in cloud so all the registered user can use the system at a single time. A lightweight process which is called as mobile host is installed in the form of application or agent on registered device. It inspects all the file activity on the system. While the user will try to do any data transfer activity e.g. downloading any file from internet, the mobile host agent will divert all the traffic to cloud via proxy server and run multiple detection engines in parallel by hosting them on emulated device. The use of virtualization to run multiple detection engines increases the coverage malware detection. The proposed system work as a proactive defense mechanism, where the Smartphone user is given the alert that the file is corrupt before it is downloaded.

While diverting the traffic to the cloud, proxy server will also function for detection of an anomaly in network. It will detect any attack in network and notify user before attack can actually occur on Smartphone and will thus prevent the attacks.

A. Experimental Setup

For Experimental analysis of proposed framework, we used Snort, which is one of the well-known open source Intrusion Detection System. The latest version of Snort was installed on Virtual Cloud. We used Smartphone Pentest Framework [9] for building and deployment of malicious apps on Android Emulator. The Android.Stels malware was used as a payload for the apps

The Android.Stels is designed to infect Android smart phones. In March 2013, Dell Secure Works released a report with details from the Dell Secure Works Counter Threat Unit's analysis of the Android.Stels Trojan. This report provides extensive details regarding the Trojan's infection vector, capabilities, network behavior, and more. It also includes a list of specific “Threat Indicators" which were useful for developing sample Snort signatures. Android.Stels Trojan can perform the following activities:

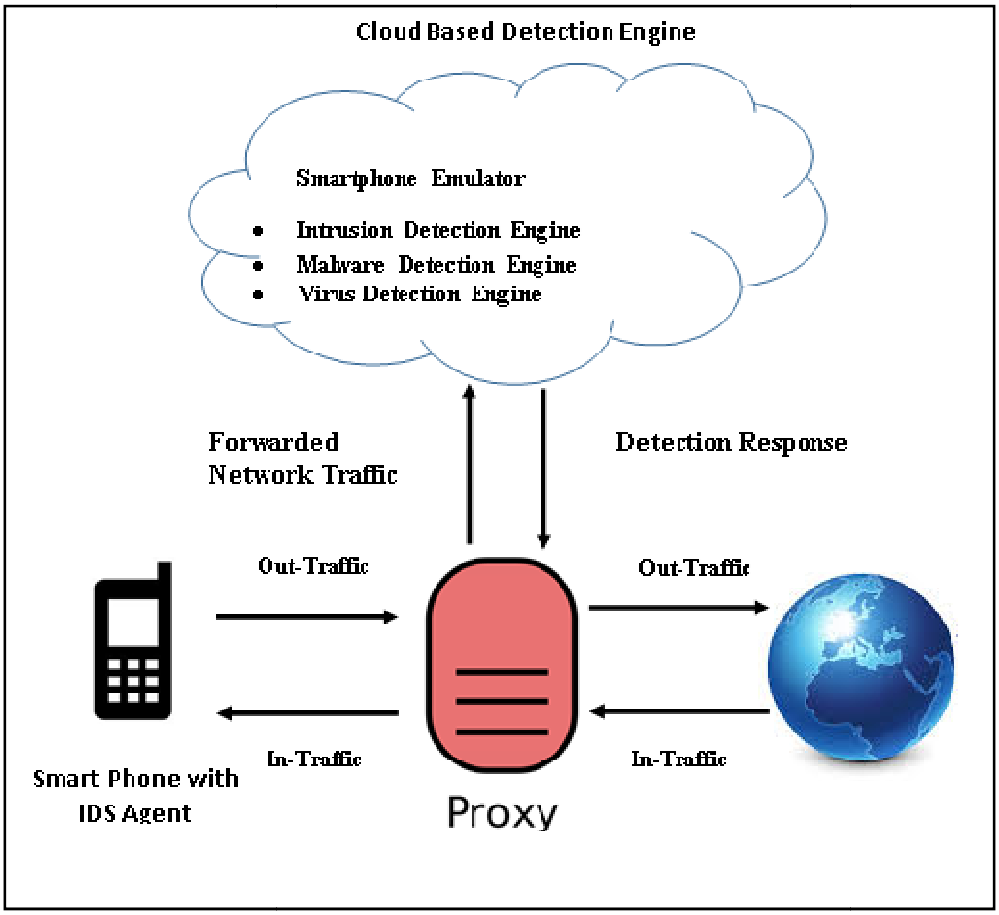

Fig 1. Cloud Based Smartphone Intrusion Detection Architecture

- Monitor SMS messages
- Selectively delete incoming SMS messages
- Send SMS messages (including to premium SMS numbers)
- Make phone calls (including to premium phone numbers)
- Send emails
- Steal the victim's contact list
- Open a web page
- Display notifications on the Android screen
- Access the phone's network settings
- Uninstall arbitrary applications
- Install arbitrary applications (including additional malware)

Based on the threat indicator released by Dell Secure Works and the preliminary analysis done by [16], set of Snort signatures were developed by [16], to detect the initial infection, as well as the subsequent Command and Control. In our experiment we used the same set of signatures developed by [16], to update our Snort signature database which is configured on virtual cloud. The details of Snort rule sets are as follows:

B. Ruleset

i.) **Command and Control Server: -**Security researchers have reported two IP addresses known to be used as Android.Stels command-and-control servers. These IP addresses are 95.211.216.148and 31.170.161.216. The following rules trigger when a packets contain the hexa decimal equivalents of these IP addresses in inbound or outbound traffic.

```
alertip any any → any
anymsg:"MOBILE_MALWARE Android/Stels
Possible CnC Server Traffic
95.211.216.148)";
content:"|5fd3d894|";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/; sid:2000007;
rev:1;)

alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
Possible CnC Server Traffic
(31.170.161.216)";
content:"|1FAAA1D8|";
classtype: trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/; sid:2000008;
rev:1;)
```

ii.) **Malicious Domains: -** Two domains are known to be associated with the Android.Stels malware. These include “ynfdb-dybdd1.freeiz.com" and “androidashplayer.net.ua.". The following rules search for these strings in the payload of IP packets.

```
alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
Malicious Domain
(ynfdbdybdd1.freeiz.com)";
content:"ynfdbdybdd1.freeiz.com";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/; sid:2000010;
rev:1;)

alertip any any -> any
any(msg:"MOBILE_MALWARE Android/Stels
Malicious Domain
(androidflashplayer.net.ua)";
content:"androidflashplayer.net.ua";
classtype:trojan-activity;
reference:url,www.secureworks.com/cybe
r-threat-intelligence/threats/stels-
android-trojan-malware-analysis/;
sid:2000009; rev:1;)
```

iii.) **Email Communications: -** Android.Stels binary contains code to send emails via HTTP POST using the domain “anonymouse.org." The following rule alerts on references to the string “anonymouse.org" within the payload of an IP packet.

```
alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
Possible Email Attempt
(anonymouse.org)";
content:"anonymouse.org";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/;
sid:2000011; rev:1;)
```

iv.) **POST Infection Process:** - Systems infected with Android.Stels “phone home” to command-and-control servers at regular intervals. The “phone home” message consists of an HTTP POST with several unique, recognizable characteristics. For example, the multipart boundary of the HTTP POST message is “AaB03x." The following Snort rule alerts on the presence of this string in the payload of an IP packet.

```
alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
```

```
POST From Infected Client";
content:"AaB03x"; classtype:trojan-
activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/; sid:2000006;
rev:1;)
```

v.) **Command and Control Commands:-** According to malware reverse engineers, the Android.Stels Trojan accepts commands from remote C&C servers, includng "RemoveAllSmsFilters," "RemoveAllCatchFilters," "SendContactList," "SendPackageList," "makeCall," and more. The following Snort rules alert on the presence of any of these commands in the payload of an IP packet.

```
alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
RemoveAllSmsFilters Command FromCnC
Server";
content:"removeAllSmsFilters";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-
android-trojan-malware-analysis/;
sid:2000001; rev:1;)

alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
RemoveAllCatchFilters Command FromCnC
Server";
content:"removeAllCatchFilters";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-
android-trojan-malware-analysis/;
sid:2000002; rev:1;)

alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
SendContactList Command FromCnC
Server"; content:"sendContactList";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/;sid:2000003;
rev:1;)

alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
SendPackageList Command FromCnC
Server"; content:"sendPackageList";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-threat-
intelligence/threats/stels-android-
trojan-malware-analysis/;sid:2000004;
rev:1;)

alertip any any -> any any
(msg:"MOBILE_MALWARE Android/Stels
makeCall Command From CnC Server";
content:"makeCall";
classtype:trojan-activity;
reference:url,
www.secureworks.com/cyber-
threat-intelligence/threats/stels-
android-trojan-malware-
analysis/;sid:2000005; rev:1;)
```

VII. RESULTS

We captured the alerts for 2 Hrs of Smartphone communication through our virtual cloud setup. During this period Snort IDS configured on cloud, successfully detected the suspicious activity triggered on Android Emulator and generated a log report. The alerts generated during this period are as follows:

```
[1:815:1] MOBILE_MALWARE Android/Stels
POST From Infected Client

[1:750:1] MOBILE_MALWARE
Android/StelsbotId Phone Home to CnC
Server

[1:683:1] MOBILE_MALWARE Android/Stels
Malicious Domain
ynfdbdybdd1.freeiz.com)

[1:650:1] MOBILE_MALWARE
Android/StelsRemoveAllSmsFilters Command
FromCnC Server

[1:569:1] MOBILE_MALWARE
Android/StelsRemoveAllSmsFilters Command
From
CnC Server
```

VIII. CONCLUSION

The objective of the research is to overcome the Smartphone resource constraints and develop an efficient IDS architecture. In this paper we proposed and developed the IDS architecture for Smartphone based on cloud. The prototype implementation demonstrated in the paper successfully detected the malicious activity as shown in the log generated by Snort. The system successfully identifies the intrusion in the Android based Smartphone using cloud. This work has to be carried out further, as there are certain limitations of Snort for detecting the intrusion in all types of protocols used in mobile communication. We either need to add some plugins in Snort to capture and analyze the various protocol used in mobile communication or find out some other alternative for

developing more robust IDS for Smartphone which is our future work.

REFERENCES

[1] Amir Houmansadr, Saman A. Zonouz, and Robin Berthier, "A Cloud-based Intrusion Detection and Response System for Mobile Phones," IEEE, 2011.

[2] Asaf Shabtai, Uri Kanonov, Yuval Elovici, Chanan Glezer, and Yael Weiss. "Andromaly: a behavioral malware detection framework for android devices". Journal of Intelligent Information Systems, pages 1–30, 2011. 10.1007/s10844-010-0148-x

[3] Aubrey-Derrick Schmidt, Hans-Gunther Schmidt, Jan Clausen, Kamer Ali Y¨uksel, Osman Kiraz, Ahmet Camtepe, and Sahin Albayrak. "Enhancing security of linux-based android devices". In Proceedings of 15th International Linux Kongress. Lehmann, October.2008.

[4] Christian Funk, Maria Garnaeva Kaspersky Security Bulletin 2013. Overall Statistics for 2013, December 10, 2013.

[5] D. Samfat and R. Molva. 2006. IDAMN: an intrusion detection architecture for mobile networks. IEEE J.Sel. A. Commun. 15, 7 (September 2006), 1373-1380.

[6] David Dagon, Xinzhou Qin, Guofei Gu, Wenke Lee, Julian Grizzard, John Levine, and Henry Owen. Honeystat: Local Worm Detection Using Honeypots. In Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID), 2004.

[7] Fangfang Yuan; Lidong Zhai; Yanan Cao; Li Guo, "Research of Intrusion Detection System on Android," Services (SERVICES), 2013 IEEE Ninth World Congress on , vol., no., pp.312,316, June 28 2013-July 3 2013.

[8] Helen J. Wang, Chuanxiong Guo, Daniel. R. Simon ,and Alf Zugenmaier. Shield: vulnerability-driven network filters for preventing known vulnerability exploits. In Proc. SIGCOMM, August 2004.

[9] https://github.com/georgiaw/Smartphone-Pentest-Framework.git/

[10] Markus Miettinen and Perttu Halonen. 2006. Host-Based Intrusion Detection for Advanced Mobile Devices. In Proceedings of the 20th International Conference on Advanced Information Networking and Applications - Volume 02 (AINA '06), Vol. 2. IEEE Computer Society, Washington, DC, USA, 72-76.

[11] N. Stakhanova, S. Basu, and W. Johnny, "A taxonomy of intrusion response systems," Computer, vol. 1, no. 1, pp. 169-184, 2007.

[12] Namitha Jacob, "Intrusion Detection in Cloud for Smart Phones," www.ijreat.org.

[13] Rohit S. Khune and J. Thangakumar "A Cloud-Based Intrusion Detection System for Android Smartphones" 2012 International Conference on Radar, Communication and Computing (ICRCC).

[14] S. Xiaonan and W. Wolfgang, "The Use of Computational Intelligence in Intrusion Detection SystemsP: A Review," Soft Computing, no. November, 2008.

[15] Sahil Sakhala, Kshitij Khakurdikar, "Anomaly Detection For Smart Phones Using Cloud-Based Intrusion Detection and Response Systems", International Journal & Magazine of Engineering, Technology, Management and Research, ISSN No: - 2320-3706, Jan 2014.

[16] Sherri Davidoff, David Harrison, Randi Price, Scott Fretheim, "Do-It-Yourself Cellular Intrusion Detection System", LMG Security, July 24, 2013.

[17] Stojan Kitanov, Danco Davcev "Mobile Cloud Computing Environment as a Support for Mobile Learning". In CLOUD COMPUTING: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, 2012, pages 99-105.

[18] Sun, B., Yu, F., Wu, K., Xiao, Y., Leung, V.: Enhancing Security Using Mobility-Based Anomaly Detection in Cellular Mobile Networks. In IEEE Trans. Vehicular Technology, 2007.

[19] Symbian white paper. Symbian smartphones for the enterprise. http://www.symbian.com/technology/smartphone enterprise.html.

[20] Vladimir B. Oliveira, Zair Abdelouahab, Denivaldo Lopes, Mario H. Santos, and Valéria P. Fernandes, " Honeypotlabsac: A Virtual Honeypot Frameworkf For Android", International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.4, July 2013.

[21] Yan, G., Eidenbenz, S., Sun, B.: Mobi-Watchdog: You Can Steal, But You Can't Run!. Proc. WiSec, 2009.