IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 12, Issue 5 (Jul. - Aug. 2013), PP 85-91
www.iosrjournals.org

Third-Party Emergency Alert Systems over Cellular Text Messaging Services Providing The Security Implications

G.Shanyora[1], A. Krishna Chaitanya[2]
[1]*M.Tech (CSE), VCE, Hyderabad India.*
[2]*Assoc.Professor.in IT, VCE, Hyderabad India.*

Abstract: *Cellular text messaging services are increasingly being relied upon to disseminate critical information during emergencies. Accordingly, a wide range of organizations including colleges and universities now partner with third-party providers that promise to improve physical security by rapidly delivering such messages. The Most of the people are stressed out and overstrained after accidents even if no one is hurt. Consequently, they may face some difficulty in reporting the accident to the police and civil defense, or they may provide them with inaccurate information about the location of the accident. Hence it will take the police and civil defense more time to reach the accident location in the appropriate time to rescue people. Our proposed system The message is sent by the affected person is by a software installed in mobile which doesn't require the user not even to type the message. Then the message will be received by the service center using GSM modem. Using the mobile number and tower location they identify accident location using latitude and longitude in Smart Phones. Then they inform the type of the accident and appropriate location to the nearby service (police & fire). This will reduce the time required for the police and the emergency personnel to reach the accident location.*

Key words: *SMS Services, Denial of Service, Reporting, Applications.*

I. Introduction

The Short Message Service (SMS) is arguably the most popular data service over cellular Networks nowadays. Though it was originally conceived as a paging mechanism for voice mails as part of the GSM specification in 1992, SMS has evolved into one of the most successful wireless data services in recent years. The SMS method of communication exhibits a number of Characteristics that have contributed to its increased popularity. First, device support for sending and receiving short messages is almost ubiquitous, ranging from low-end mo- bile phones to Web-interface gateways that can be accessed via ordinary PCs over the Internet.

Moreover, delivery and routing of SMS messages are supported by most cellular networks around the world. Second, it follows a "push" model of operation and short messages are delivered to mobile de- vices in near real-time, making SMS the wireless counterpart of Internet-like instant messaging applications such as ICQ and AIM. At present, many manufactures have also developed their own SMS alert system to send immediately an SMS message to the responsible people when their devices are in trouble or some abnormal condition is detected. For a system which consists of devices from many different manufactures, the integration of these manufactures' SMS alert systems is difficult.

Accordingly, SMS messaging is now viewed by many as a reliable method of communication when all other means appear unavailable. In response to this perception, a number of companies offer SMS-based emergency messaging services. Touted as able to deliver critical information colleges, universities, and even municipalities hoping to coordinate and protect the physical security of the general public have spent tens of millions of dollars to install such systems. Unfortunately, these products will not work as advertised and provide a false sense of security to their users. The fluidity of the markets also presents enormous security challenges.

Rapidly developed and deployed applications coarse permission systems privacy invading behaviors malware and limited security models has led to exploitable phones and applications. Vulnerable applications will find their way to market. In this paper, we broadly characterize the security of applications in the Android Market. In contrast to past studies with narrower foci, e.g., we consider a breadth of concerns including both dangerous functionality and vulnerabilities, and apply a wide range of analysis techniques.

In this paper, we explore the limitations of third-party Emergency Alert Systems (EAS). The rest of this paper presents our argument in more detail. Section 2 Related Works. Section 3 System design and architecture, and Section 4 Android applications J2EE, GSM modem and GPS. Section 5 summarizes our results and discusses future work.

II. Related Work

Internet Denial of Service:

Most of the evaluation of DoS solutions has focused on their feasibility and/or effectiveness in preventing DoS attacks. While this is clearly important, we also draw attention to a different kind of issue with existing work in this field: the future consequences for the Internet and its applications if these approaches are adopted. For example, some low-end routers will crash if they are sent pings at too fast a rate, because their CPU becomes overwhelmed. While this might seem to be easily fixed by building more robust software, as a practical matter, almost every device connected to the Internet has some vulnerability to a flooding attack, just as almost every host on the Internet is vulnerable to a virus attack. Our experience with new applications on Planet Lab indicates that a good DoS solution must not only be effective, it must also permit the seamless introduction of new network services.

Nationwide Short Message Service:

This measurement-based study focuses on the reliability of the SMS service during both normal and overload operating conditions. Generally speaking, reliability denotes the ability of a system to consistently provide service with certain performance characteristics, even in the presence of stressful conditions that may threaten to disrupt the offered service. It may take several forms and includes notions such as high availability, resiliency, fault tolerance, and even security, depending on the context. In this work, we first seek to establish a baseline characterization of the reliability for end-to-end short message transfer during normal traffic conditions.

We then examine reliability during stress conditions of “flash-crowd” events that are frequently observed during special occasions such as the New Year’s Eve. Two factors that may critically affect reliability of SMS in the near future are further investigated. The first is bulk message delivery which has been increasingly employed by commercial entities to reach mass markets. It may incur heavy network congestion and reduce the message delivery quality. The second is the topological structure of the social network formed by the SMS users, which can be exploited to allow fast spreading of security attacks such as viruses and spams. To the best of our knowledge, this is the first study on SMS reliability that is based on measurements from a real, operational cellular carrier.

Address Filtering:

This would prevent attackers from placing arbitrary source addresses in their packets and would therefore be useful in reducing the kinds of attacks that could be launched. Source filtering can be generalized to allow for filtering any packet that cannot have legitimately arrived at any point in the middle of the network [16].

Ingress filtering becomes effective only with a high degree of deployment; a source address only provides proof of authorship if every node in the network is part of the trusted computing base. Despite being recommended as a “best practice” for over five years, there are still many gaps in the enforcement of ingress filtering. Even with complete deployment, advances in attack methods have largely rendered source filtering irrelevant. Source addresses can be spoofed among all addresses sharing the same network prefix behind the filter; this can be thousands of addresses. Worse, automated virus tools have made it easy to enlist very large numbers of hosts in a given attack; attacks today often comprise lots of legitimate, un spoofed packets – if a million machines send your DSL modem a single TCP SYN packet, it doesn’t matter that they are all using their real source address your link will be unusable.

III. System Design and Architecture

SMS Network Architecture:

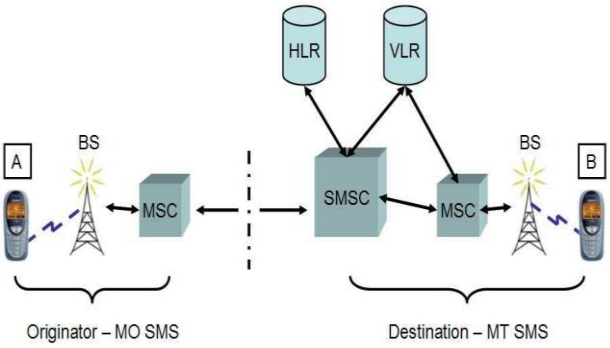

Figure 1 SMS Service Architecture

In current cellular networks, SMS messages are transmitted over the Common Channel Signaling System 7 which is the digital signaling control network used by net- work elements of wire line and wireless telephone carriers to exchange control information for call setup, routing, mobility management, etc. Figure 1 shows the typical network elements and architecture employed for handset-to-handset communication (Point-to-Point SMS). Conceptually, the network architecture is split into two segments that are central to the SMS philosophy of operations. The elements at the sender, i.e., the Mobile Originating (MO) part, include the Mobile Station (MS) of the sender, the Base Station (BS) that provides the radio infrastructure for wireless communications, and the Originating Mobile Switching Center (MSC) that manages routes and switches all traffic into and out of the cellular system on behalf of the mobile device of the sender. The elements at the destination of the message, more often known as the Mobile Terminating (MT) part, also feature a base station and an MSC (Terminating MSC) for the receiver. In addition, an SMS Center (SMSC) acts as a centralized, store and forward server that is responsible for accepting, storing, retrieving subscriber information, and forwarding messages to the intended recipients of the messages. It is assisted by two databases, namely the Home Location Register (HLR) and the Visitor Location Register (VLR), in which location information is kept regarding the subscribers and their mobile devices (such as the address of the MSC that the device is associated with). For more details on the SMS network architecture and its operations, as well as other scenarios with fixed entities that are capable of sending and receiving short messages (e.g., application servers),Figure 2 we refer the interested reader to the tutorial and the specification.

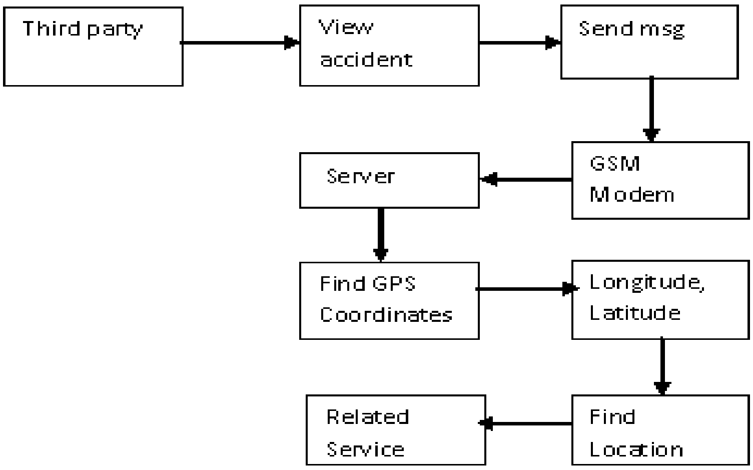

Figure 2 Accident Reporting Services Block diagram

About J2ME Mobile Applications:

J2ME (Java 2 Micro Edition) is a Java platform from Sun Microsystems that allows programmers to use the Java programming language and related tools to develop programs for small devices with a limited processor power and small memory size like mobile phones, personal digital assistants etc. A configuration defines the minimum set of JVM (Java Virtual Machine) features and Java class libraries available for a particular category of devices. A configuration typically represents a group of devices with similar processing power and amounts of available memory. A profile is a specification that defines sets of APIs and features and utilizes the underlying configuration to provide a complete run-time environment for a specific kind of device. The profile manages the application, user interface, networking and I/O.

J2ME defines two configurations, the CLDC (Connected Limited Device Configuration) and the CDC (Connected Device Configuration). The CLDC is for devices with small amounts of memory and/or slow processors. The VM (Virtual Machine) used by the CLDC omits some important features. The CDC, on the other hand, includes a full JVM and a much larger set of core classes, so it requires more memory than the CLDC and a faster processor.

IV. Performance and Results

Figure 3 Android Smart Phones

The rapid growth of smart phones has lead to a renaissance for mobile services. Go-anywhere applications support a wide array of social, financial, and enterprise services for any user with a cellular data plan. Application markets such as Apple's App Store and Google's Android Market provide point and click access to hundreds of thousands of paid and free applications (Figure 3). Markets streamline software marketing, installation, and update herein creating low barriers to bring applications to market, and even lower barriers for users to obtain and use them. The fluidity of the markets also presents enormous security challenges. Rapidly developed and deployed applications coarse permission systems privacy invading behaviors malware and limited security models have led to exploitable phones and applications.

In this paper, we broadly characterize the security of SMS Services Applications in the Android Market. In contrast to past studies with narrower foci we consider a breadth of concerns including both dangerous functionality and vulnerabilities, and apply a wide range of analysis techniques. In this, we make two primary contributions:

- We design and implement a Dalvik decompilier, ded. Ded recovers an application's Java source solely from its installation image by inferring lost types, performing DVM-to-JVM byte code retargeting, and translating class and method structures.
- We analyze 21 million LOC retrieved from the top 1,100 free applications in the Android Market using automated tests and manual inspection. Where possible, we identify root causes and posit the severity of discovered vulnerabilities.

The Android system architecture:

Android is an OS designed for smart phones.

Depicted Android provides a sandboxed application execution environment. A customized embedded Linux system interacts with the phone hardware and an off-processor cellular radio. The Binder middleware and application (Figure 5) API runs on top of Linux. To simplify, an application's only interface to the phone is through these APIs. Each application is executed within a Dalvik Virtual Machine (DVM) running under a unique UNIX uid.

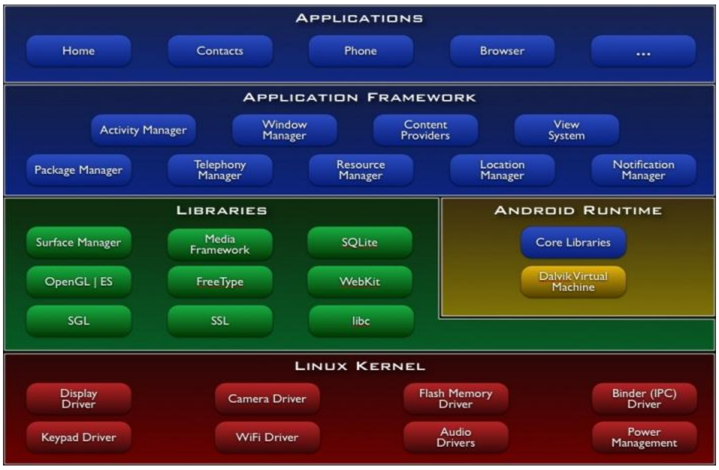

Figure 4 Android Architecture

Dalvik Virtual Machine:

Java applications are composed of one or more .class files, one file per class. The JVM loads the byte code for a Java class from the associated .class file as it is referenced at run time. Conversely, a Dalvik application consists of a single .dex file containing all application classes. Figure 2 provides a conceptual view of the compilation process for DVM applications. After the Java compiler creates JVM byte code, the Dalvik dx compiler consumes the .class files, recompiles them to Dalvik byte code, and writes the resulting application into a single .dex file.

This process consists of the translation, reconstruction, and interpretation of three basic elements of the application: the constant pools, the class definitions, and the data segment. A constant pool describes, not surprisingly, the constants used by a class. This includes, among other items, references to other classes, method names, and numerical constants. The class definitions consist in the basic information such as access flags and class names. The data element contains the method code executed by the target VM, as well as other information related to methods (e.g., number of DVM registers used, local variable table, and operand stack sizes) and to class and instance variables.

Client: The mobile requests its location from the positioning system periodically and sends it through the communication network to the server, (Who are near in accident).A location based service is an information service that can be accessed using the mobile device through the mobile network and utilizes the ability to make use of geographical positions of the mobile device. The user can request the location of a Accident place.

Client Application:

The proposed approach supports concurrent access of remote database by multiple J2ME applications. The J2ME android application can access the remote database either through a J2EE web application or J2SE SMS application. Both the J2EE web application and J2SE SMS application reside on remote server along with the database. While internet GPRS and GPS connectivity is available, J2ME android application uses J2EE web application to access database and uses J2SE SMS application while internet connectivity is unavailable. For communicating with J2EE web application, our J2ME client application uses HTTP which is supported by all MIDP devices and GPRS. HTTP makes wireless applications more portable and standard. For coding purpose; packages like java.io, javax.microedition.io have been used. A small portion of J2ME code is shown which has been used to send and receive data with HTTP.

Server: The server receives users alert message after receiving the message, the server side that is Traffic Monitoring Center(TMC) check the what type of message it is any emergency ,after checking make to find an corresponding location of the accident place through GPS, and after founding the location to make an call to the corresponding service. The map will be showing on server application in current position.

J2EE (Server) web application:

After receiving queries from J2ME client application, our J2EE web application uses Apache tomcat server, JDBC driver and JDBC API to process those queries accessing the database. After processing queries, J2EE web application sends back a response containing data to J2ME client application. The package java.sql has been used. Some used classes are Statement, Result set, Connection, Driver Manager, Data Request etc. The accident place maps showing in J2EE web application (Figure 6) are built on JSP (Java Server Page), Servlet etc.

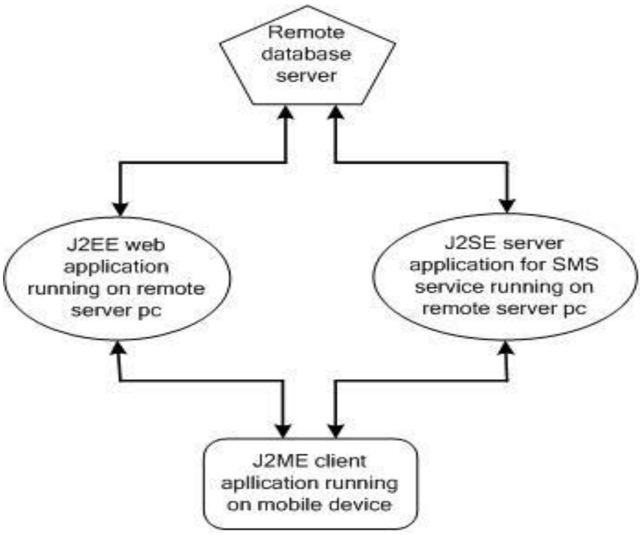
Remote database server
J2EE web application running on remote server pc
J2SE server application for SMS service running on remote server pc
J2ME client apllication running on mobile device

Figure 5 Architecture of the J2ME and J2EE

Gps:

The Global Positioning System (GPS) is a space-based satellite navigation system that provides location and time information in all weather conditions, anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites. The system provides critical capabilities to military, civil and commercial users around the world. It is maintained by the United States government and is freely accessible to anyone with a GPS receiver. Every time the mobile phone updates the user location in the server, it requests the location of the user from the GPS. The GPS determines the longitude and the latitude (geographical) and sends them to the mobile phone.

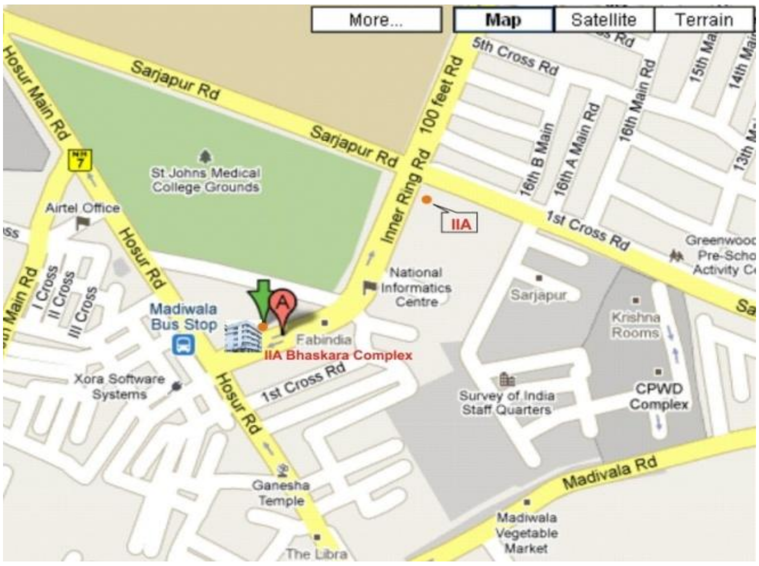

Figure 6 Server Displaying Reporting Place

GSM and Database:

The database contains all users subscribed in the service with their receiving message, (received message will stored in database) after receiving the message, we are finding the location of the corresponding accident location. GSM modem is a specialized type of modem which accepts a SIM card, and operates over a subscription to a mobile operator, just like a mobile phone. From the mobile operator respective, a GSM modem looks just like a mobile phone.

Using GSM Modem:

Importing the comm., Driver and connecting the Modem to the PC with serial port. Then the application uses standard JDBC driver to access database based upon those queries. After query execution, J2SE SMS Server application sends back response data back to J2ME client application via a SMS through the GPRS modem. At coding level we use Java Communication API. The used packages are javax.comm.CommPort, javax.comm.CommPort CommPortIdentifier, jvax.comm.SerialPort etc. A portion of J2SE code of SMS application is shown bellow.

```
CommPort the Port;
CommPortidentifier thePortID;
ThrPort = thePortID.open (strPortName,
TIMEOUTSECONDS*1000);
Serial Port my Port = (Serial Port) the Port;
Is = new BufferdReader (new
InputStreamReader (thePort.getInputStream
```

Comm Port is an abstract class that describes a communications port made available by the underlying system. Comm Port Identifier is the central class for controlling access to communications ports. Serial Port is a RS-232 serial communications port. It describes the low level interface to a serial communications port made available by the underlying system.

V. Conclusions

J2ME database access has been a very important topic for the J2ME android application developers. The proposed approach deals with some major points in J2ME SMS services. The proposed approach will help J2ME developers to use database for their J2ME android applications as well as to make distributed mobile applications with J2ME which will have a great impact on business perspective also. This proposed approach can be further updated with more features regarding J2ME database access.

References

[1]. Daniel Adkins, Karthik Lakshminarayanan, Adrian Perrig, and Ion Stoica. Taming IP packet flooding attacks. In Proceedings of Workshop on Hot Topics in Networks (HotNets-II), November 2003.

[2]. Micah Adler. Tradeoffs in probabilistic packet marking for IP traceback. In Proceedings of 34th ACM Symposium on Theory of Computing (STOC), 2002.

[3]. D. Scott Alexander, Kostas G. Anagnostakis, William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. The price of safety in an active network. In SIGCOMM '99, 1999.

[4]. David G. Andersen. Mayday: Distributed filtering for Internet services. In Proceedings of USITS, 2003.

[5]. Tom Anderson, Timothy Roscoe, and David Wetherall. Preventing Internet denial-of-service with capabilities. In Proceedings of Hotnets-II, November 2003.

[6]. S. Bellovin, M. Leech, and T. Taylor. The ICMP traceback message. Internet-Draft, draft-ietf-itrace-01.txt, October 2001.

[7]. Robert Braden. Requirements for Internet hosts – communication layers. Internet Request for Comment RFC 1122, Internet Engineering Task Force, October 1989.

[8]. T. Anderson, T. Roscoe, and D. Wetherall, "Preventing Internet Denial of Service with Capabilities," Proc. ACM Workshop Hot Topics in Networking (Hot Nets), 2003.

[9]. K. Argyraki and D.R. Cheriton, "Scalable Network-Layer Defense against Internet Bandwidth-Flooding Attacks," ACM/IEEE Trans.Networking, vol. 17, no. 4, pp. 1284-1297, Aug. 2009.

[10]. Associated Press, "Man Admits Sending 'Monkey Out of Cage' Message,"

[11]. http://www.google.com/hostednews/ap/article/ALeqM5gjBi_YGzVmUqV0YDKifMv, 2009.

[12]. S. Blons, "Emergency Team Aids Efforts," http://graphic. pepperdine.edu/special/2007-10-24-emergencyteam.htm, 2007.

[13]. M. Casado, P. Cao, A. Akella, and N. Provos, "Flow Cookies: Using Bandwidth Amplification to Defend against DDoS Flooding Attacks," Proc. Int'l Workshop Quality of Service (IWQoS), 2006.

[14]. Cellular-News, "Malaysian Operators Dismiss Hoax SMS," http://www.cellular-news.com/story/31247.php, 2008.

[15]. T. Christensen, "Ga. Tech Building Cleared after Blast," http://www.11alive.com/life/pets/story.aspx?storyid=106112, 2007.

[16]. CollegeSafetyNet.com, http://www.collegesafetynet.com, 2008.

[17]. Courant.com, "University Emergency SMS Service Doesn't Deliver," http://www.courant.com, Nov. 2007.

[18]. B.K. Daly, "Wireless Alert & Warning Workshop,"http://www.oes.ca.gov/WebPage/oeswebsite.nsf/ClientOESFileLibrary/ Wirel, 2011.

[19]. e2Campus, "Mass Notification Systems for College, University & Higher Education Schools by e2Campus: Info on the Go!" http://www.e2campus.com, 2008.