INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Two Way Authentication Protocol for Android Based Applications

Subhasis Banerjee
Professor, Department Computer & System Sciences, Visva-Bharati, Santiniketan, West Bengal, India
Utpal Roy
Professor, Department Computer & System Sciences, Visva-Bharati, Santiniketan, West Bengal, India

Abstract:
In this paper, we present a scheme to authenticate SMS communication. In the suggested scheme authentication text is generated depending upon a ticket. The ticket is generated randomly by the service provider. Our authentication scheme is meant to work on android platform. The Android platform has been dealt as a topic of mobile security because Android is an open platform whose sources can be observed by anyone. Unauthenticated message may create serious problem for many mobile based applications.. In this paper, we have developed a two way authentication system which can work for any mobile based applications in android. We have also used simple hashing technique to create the message digest.

Keywords: *BTS, UE, SMSC, PKI*

1. Introduction

1.1. Short Message Service (SMS)

SMS stands for short message service. It is a method of communication that sends text between cell phones, or from a PC or hand-held to a cell phone. The maximum size of the text messages is: 160 characters (letters, numbers or symbols in the Latin alphabet). For other alphabets, such as Chinese, the maximum SMS size is 70 characters

1.2. Need for Authenticated Message Transmission

Information security means protecting information. It secures information systems from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction. Maintaining privacy and authentication in our personal communication is something everyone desires. As short message service (SMS) is now widely used for official communication its security and authenticity has become a major concern for business organization and customers. So there is a need for authenticated SMS to check its validity.

1.3. Working of SMS

It is well known that SMS service[1] is a cell phone feature but indeed, SMS can also work on other computing devices now a days , such as PC, Laptop, or Tablet PC as long as they can accept SIM Card. SIM Card is needed because SMS service needs SMS center client which is built in on the SIM Card.

1.3.1. BTS

A base transceiver station (BTS) is a piece of equipment that facilitates wireless communication between user equipment (UE) and a network. UEs are devices like mobile phones (handsets), WLL phones, computers with wireless internet connectivity, WiFi and WiMAX devices and others.

1.3.2. MSC

The mobile switching centre (MSC) is the primary service delivery node for GSM/CDMA, responsible for routing voice calls and SMS as well as other services (such as conference calls, FAX and circuit switched data). The MSC sets up and releases connection. This end to end connection handles mobility and hand over requirements during the call and takes care of real time prepaid account monitoring etc...

1.3.3. SMSC

When SMS is transmitted from a cell phone, the message will be received by mobile carrier's SMS Center (SMSC), [2] which does destination finding, and then send it to destination devices (cell phone). SMSC is SMS service centre which is installed on mobile carrier core networks. Beside as SMS forwarding, SMSC also acts as temporary storage for SMS messages. So, if the destination cell phone is not active, SMSC will store the message and then deliver it after the destination cell phone is active. As additional, SMSC also notify the sender whether the SMS delivering is success or not. However SMSC cannot store the SMS message forever since the storage capacity is not unlimited. During the SMS delivering, sender cell phone and SMSC is actively communicating. So, if the non active destination cell phones become active, SMSC directly notifies the sender cell phone and tell that the SMS delivering is success. This is how the SMS works in general.

So we can say that after sending of SMS, the SMS Center (SMSC) is used to store the SMS messages in order to forward them to the target mobile device. SMSC uses Store-and-forward technique to store messages in order to forward to the target device. If the (Home Location Register) HLR of target mobile device is active, then SMSC will transfer the SMS message to target mobile device. SMSC will receive the verification message that confirms the delivery of SMS message to target device. Unencrypted SMS messages are stored in SMSC; therefore, SMSC staff can view and modify the content of SMS message. Many SMSCs can also keep the copy of SMS message for billing and auditing purposes. Therefore, it becomes easy for attackers to view SMS messages through SMSC. After attacking SMSC, attacker can read the SMS messages. Several Cryptography methods have been used to reduce the SMS security threats and provide enough security to mobile devices. But these encryption techniques can't perform their activity in a complete manner since it affects the performance of mobile devices in terms of power and battery life constraints. Symmetric Cryptography is the type of encryption used to provide end-to-end security to SMS messages. It is also good for mobile devices due to their limited resources, i.e., limited power/energy, in-sufficient memory and less processing power. It uses the shared secret key between two parties in order to protect SMS message communication. Key distribution mechanism remains in-secure, since if an attacker intercepts the key distribution process and intercepting the key, he/she can easily modify the SMS message contents. Therefore, Key distribution is quite difficult and insecure in symmetric key cryptography. DES and AES are the examples of symmetric key cryptography. The key distribution problem is solved by Asymmetric cryptography by using pair of keys (i.e. private and public) for communication. Sender is using public key for communication while private key is used in order to decrypt the message. Man-in-the-middle attack is common in public key cryptography. Public key infrastructure (PKI) is then used to improve the deficiency of public key cryptography. Although Asymmetric encryption is strong and key distribution is also very easy in it, but, it is avoided because of its computational overhead.

Nevertheless, mobile devices have improved their memory capacity as well as their performance. Energy efficiency and battery technology is also improved in order to extend the operational time of mobile devices. Besides of these developments, it is still a research question that whether symmetric and asymmetric encryption can fully provide their advantages to secure mobile SMS messages.

2. Literature Survey

There are many studies reported on the security and privacy-preservation issues for VANETs [2]–[11]. To achieve message authentication Raya and Hubaux [5] proposed that each vehicle should be pre-loaded with a large number of anonymous public and private key pairs together with the corresponding public key certificates. Traffic messages are signed with a public-key-based scheme. To achieve privacy, each public and private key pair has a short life time, and a pseudo ID is used in each public key certificate. This scheme requires a large storage capacity to store this security information and that is why it not suitable for VANET.

Lin et al. [6] proposed a group-signature-based scheme to sign each message. Since there is no identity information included in messages, this approach can also achieve identity privacy preservation. Furthermore, the group-signature-based scheme reduces the storage cost.

Calandriello et al. [7] developed a similar scheme to reduce the overhead of the group-signature-based scheme and in this scheme vehicle can generate public and private key pairs by itself by using a group key. This scheme can achieve a tradeoff between the group-signature-based scheme and the traditional PKI-based scheme.

Xi et al. [8] introduced a random key-set-based authentication protocol to preserve the vehicles' privacy.

3. Android Architecture and Security Related Issues

The Android platform is the most popular platform for mobile devices. It is designed in a way that applications can be easily installed through online application markets. In order to secure such a system, Android implements a security concept [9] [10] which protects the system as well as the applications by isolating the application context and permission Management. Android is not a single piece of hardware; it's a complete, end-to-end software platform that can be adapted to work on any number of hardware configurations. Everything is there, from the boot loader all the way up to the Applications. In the following subsection we discuss the Android architecture.

3.1. Architecture

The architecture of Android is based on a Linux Kernel and adopts many concepts, which are common on UNIX systems. Android uses different system users and applies file system permissions which isolate applications. In order to allow communication and interaction beyond this isolation, Android provides interfaces, which are protected by a permission management.

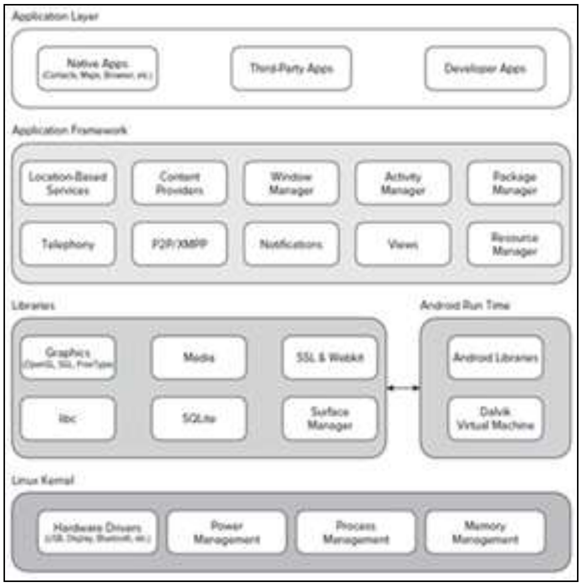

Figure 1

The Android architecture consists of five main component groups, as shown in figure- 1. The underlying Linux kernel manages any hardware access and provides interfaces for device specific drivers, which were provided by hardware vendors. The configuration of the kernel on Android has been adjusted specially for mobile devices. Therefore, many features, which are enabled on desktop and server systems, have been disabled in order to reduce memory and energy consumption. The second component group contains native libraries, which provide different services e.g. for graphic rendering, cryptographic functions and database access etc.

These components are executed natively and can be used within Android applications by loading them into their process context. So, these components are stored as shared objects. Such a process of an application also contains a copy of the Dalvik Virtual Machine (DVM), as part of the third component group shown in figure 1. The DVM executes Android applications, which are delivered in form of Dalvik byte code. Within the DVM, an application can make use of services, which are provided by the Android Application Framework and native libraries. In this way the Android Application Framework can ensure compatibility between different systems and also restrict data access by permission checking.

3.2. Security

Android is a multi-process system, in which each application (and parts of the system) runs in its own process. Most security between applications and the system is enforced at the process level through standard Linux facilities, such as user and group IDs that are assigned to applications.

3.3. SMS Applications with Android

Now a days SMS is one of the most used application on mobile phones. It is very useful for small person to person message communications. It can also be used in a variety of ways within applications beyond its normal use. For instance, it can be used to exchange small amounts of data between phones running the same application. With most built-in functionality in Android we can use SMS message, utilizing the normal SMS Activity. We can also develop an application which receives SMS messages. In this type situation we need to tell Android that we want to handle incoming SMS messages. SMS handling in Android is managed by launching a specific class when an SMS comes in. Android Application Framework can restrict data access by permission checking. But users are not offered any fine-tuned control of what permissions to grant the applications. If an application requests for access or permissions the user are not comfortable with granting, so the only choice is to not install the application. There are suggestions of improvement to the permission system, such as the ability to grant only a subset of requested permissions, and also advocate for the inclusion of Security Enhanced Linux (SELinux) to further protect the Linux kernel using Mandatory Access Control policies.

Access control is a very useful measure against many types of attacks. The rapid increase of malware on Android can compromise a device and steal sensitive information. Most of these malware comes in the form of Trojans which provide a method to inject arbitrary code into application without invalidating their signature. So integrity of any data or message must be kept intact to save the system from Trojans. There was a larger security analysis of Android carried out by Berger et al [11]. They found discrepancies between the documentation of Bluetooth communication available to developers and the actual implementation in the Android system itself. Several weaknesses are identified regarding communication, both between applications and between an app and the Internet. Since Google does not use a manual review process it is important that the tools used to automatically review submitted applications can have weak configurations. This type of drawbacks will make Android based communication channels vulnerable.

So in this situation we must ensure the SMS communication to be authentic. We have to keep in mind that SMS is used not only for personal communication now a days but it is also used for secure system messages[12][13], secure e-commerce communications etc.

4. Working Procedure of Our Two Way Authentication

In our approach we have used Transposition cipher combined with simple substitution. The parameters of this process is chosen depending upon the token supplied by the service provider. The sequential steps of the approach is as follows:

1. In this mechanism service provider **randomly** generates a token in a given range of values.
2. The service provider encrypts the token with the sender's public key and send it to sender.
3. The service provider also encrypts the token with the receivers public key and send it to receiver
4. The sender and the receiver decrypts the token with their respective private key.
5. The token number indicates the parameters to be chosen for authentication purpose.
6. The code for authentication is IEMI number followed by the hash code of the current message.
7. Now both sides can attach their code below the message.
8. Both side sends the composed message to each other and now authentication of the messages of both sides can be verified.

Both sides continue or close their session as per their requirement

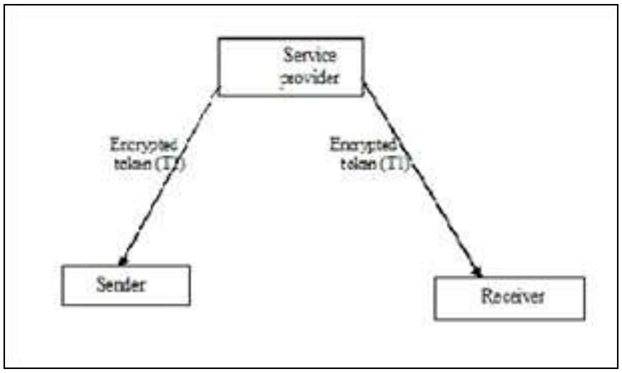

Figure 2

The confidentiality of the message can be ensured by our suggested procedure on confidentiality [14]. So by combining these two we will get a message which is confidential and authentic.

These type of messages will be very much useful in VANET scenario to share congestion and other status in the road. Other vehicular nodes will fail to penetrate this communication because their authentication check will fail. Depending upon the parameter values in the message, the vehicles will choose their route. The vehicle in the common locality of interest can take part in the SMS exchange. As per our design total system will work on android based systems. The communication network should be provided by a standard mobile service provider. Android is chosen because of ease of development.

5. Performance Analysis

Small messaging is used by huge number of users for both private and business communication and is a low-cost option for mobile devices. But most of the mobile messaging applications do not provide end-to-end security. We would like to use our authentication technique for messaging in vehicular network. As a supporting platform we have taken Android because it is basically a Linux based system. So by default it is secure. It is also very light to load on a mobile device and it is very fast compared to other platforms. It is also less prone to male wares compared to other platforms

In Android, every user has six different options to choose from lock screen, all of which offer their levels of security. Our message authentication procedure will provide an extra level of security on the top of it. The implementation of our authentication Technique is under progress and after testing the applications from various angles finally it will be place in the Google play store for free public use. It is anticipated that this application will be equally powerful and user friendly like other authentication techniques as available in the web.

6. Conclusions

The SMS services are inevitably in mobile banking, e-commerce, and defense applications and in many others. At the time of official use of SMS we not only want to get it in securely but we also want it to be authentic. Unless a message is authentic there is no validity of the message. Authenticity becomes more important when we use it for message communication in ad hoc network such as VANET. Considering the issue from the computational point view, low complexity is always preferred in a low power driven Android based system. In our proposed scheme we have chosen a private key crypto-system which is cost effective considering the computational complexity. Furthermore we check the integrity of the message by the use of hash function. Our random selection of key from a key spool is very good technique to increase the overall security level of the system.

7. Acknowledgment

The authors acknowledge the continued interest of the faculty members of the department on the Android based authentication technique, the subject of discussion of the paper.

8. References

i. M. Toorani and A. A. Beheshti Shirzai, SSMS . A Secure SMS Messaging Protocol for the M-Payment Systems, IEEE Symposium on Computers and Communications, 2012, 700-705

ii. U.S. Dept. Transp., Nat. Highway Traffic Safety Admin.,Vehicle Safety Communications Project, 2006. Final Rep.

iii. S. Lee, G. Pan, J. Park, M. Gerla, and S. Lu, "Secure incentives for commercial ad dissemination in vehicular networks," in Proc. ACM Int. Symp. MobiHoc, 2007, pp. 150 – 159.

iv. Dedicated Short Range Communications (DSRC). [Online]. Available: http://grouper.ieee.org/groups/scc32/dsrc/index.html

v. M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," J. Comput. Secur. , vol. 15, no. 1, pp. 39–68, Jan. 2007.

vi. X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: A secure and privacy-preserving protocol for vehicular communications," IEEE Trans. Veh. Technol., vol. 56, no. 6, pp. 3442–3456, Nov. 2007.

vii. G. Calandriello, P. Papadimitratos, A. Lioy, and J.-P. Hubaux, "Efficient and robust pseudonymous authentication in VANET," in Proc. Int. Work-shop VANET, 2007, pp. 19–28.

viii. Y. Xi, K. Sha, W. Shi, L. Schwiebert, and T. Zhang, "Enforcing privacy using symmetric random key-set in vehicular networks," in Proc. ISADS, 2007, pp. 344–351.

ix. Adrienne Porter Felt, "Android Permissions: User Attention, Comprehension, and Behavior" , Symposium on Usable Privacy and Security (SOUPS) 2012, July 11-13,2012, Washington, DC, USA http://source.android.com/posts/opensource

x. Berger B.J., Bunke M., and Sohr K., An Android Security Case Study with Bauhaus, Working Conference on Reverse Engineering, 179–183 (2011)

xi. K. Ren, W. Lou, R. H. Deng, and K. Kim, "A novel privacy preserving authentication and access control scheme in pervasive computing environments," IEEE Trans. Veh. Technol., vol. 55, no. 4, pp. 1373- 1384,Jul. 2006.

xii. A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The TESLA broadcast authentication protocol," RSA Crypto., vol. 5, no. 2, pp. 2–13,2002.

xiii. Subhasis Banerjee ,Utpal Roy, " Secure SMS Communication in Android based System with Two Stage Protection", International Journal of Computer Science and Mobile Computing, Vol.4 Issue.6, June- 2015, pg. 1057-1064