ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

Personalized Smartphone Search Engine Enhanced Security Using MAC Technique

Roly Pandey

Pune Institute of Computer Technology, University of Pune, Pune (MH) India

ABSTRACT: Android is an open source mobile operating system developed by the android open source project (ASOP) developed by Google, use of mobile operating system is becoming widely popular, and is being used more and more these days by security personnel such as the department of homeland security, firefighters and also the emergency medical services etc which demands the cell phone to provide robust security. The idea of the project is to create personalized mobile search engine (PMSE) and provide security for user profile by using remote lock and wipe system with integrity checking of SMS notification using MAC based technique which uses a short piece of information which is used to prove the authenticity and the integrity of the message . Integrity determines accidental and intentional message changes and authenticity determines the messages origin .

KEYWORDS-: Clickthrough data , concept, location search, mobile search engine, ontology, personalization

I. INTRODUCTION

Android is an open , security focused mobile platform that is programmed with java . Android combines os features like efficient shared memory, preemptive multitasking , unix user identifiers (uid) and file permissions with the type safe java language and its familiar class library . The resulting security model is much more like a multiuser server than the sandbox found on the J2ME or blackberry platforms . unlike in a desktop computer environment where a user applications all run on the same UID.
The Aim of the our project is create personalized mobile search engine (PMSE) and provide security for user profile by using remote lock and wipe system with integrity checking of SMS notification using MAC based technique which uses a short piece of information which is used to prove the authenticity and the integrity of the message . Integrity determines accidental and intentional message changes and authenticity determines the messages origin

II. PRIVACY RISKS IN ANDROID

To explain the design of the privacy controls several initial measurements and analysis of today's android applications have been performed .As an application cannot misappropriate data it does not have access to the frequency with which applications request access to reach type of potentially sensitive data was studied then determined the frequency with applications exfiltrate data of each type and where they send the data to.
Misappropriation: Prior work has revealed that some Android applications do exploit user data for purposes that may not be expected nor desired by users. Enck et al., who developed the TaintDroid information-flow tracking system extended in our work, used this system to analyze 30 Android applications that required access to the Internet and either users' location, camera, or microphone. They found that half of these applications shared users' locations with advertisement servers. The problem is not unique to Android. Egele et al. used static analysis to track information flow in popular iPhone applications and discovered that many contained code to send out the unique device ID. Smith captured network traffic to observe iPhone applications transmitting device IDs. The Wall Street Journal commissioned its own study of 50 iPhone applications and 50 Android applications, also using a network-observation approach . The article suspects that these unique IDs are so commonly transmitted because they can be used to profile users' behaviors across applications.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

III. PRIVACY CONTROL

Remote Lock/wipe data: The remote lock and wipe system consist of a remote control module on a server and a command handling module on a smartphone. The commands are sent by sms push notification message. For example , when the user send a lock command to the smartphone via the remote control module , the remote handling module enables the password locking function to lock the smartphone. Similarly , by sending a wipe command , all personal data such as smart wallet , smart keys, contacts, SMS, Email, photos, and movie clips are remotely deleted.

Data shadowing: Data shadowing is a method for regulating privacy control since today's applications do not suspect the use of shadowing, .suggestion for simple shadow data rather than developing more elaborate uses to fool applications that might attempt to detect shadowing has been made. However, the implementation can be easily extended to support more sophisticated shadow data than what is presented below if it becomes necessary to do so. Android applications use the file system to access the camera, microphone, and logs. When applications try to open these resources, an illusion of opening an empty file is provided. Similarly, we shadowed browser metadata (history and bookmarks), SMS/MMS messages, subscribed feeds, contacts, accounts, and calendar entries by returning an empty set of data. When applications request the device's location, we return the coordinates 37.421265, -122.084026. When applications request the device's phone state, we construct phone state with a fixed phone number (1 650 623 4000) and an application-specific device ID. The shadow de- vice ID (IMEI) is generated by hashing a three-tuple containing the device ID, application name, and a secret salt randomly generated for the device. The salt ensures that an application that is granted access to the device ID cannot be linked to an application that is granted access to the shadow ID. The result of the hash is a string containing 15 decimal digits—the proper format for a GSM IMEI number. The Android phone state permission also grants access the software version number (IMEI/SV), SIM serial number, voice mail number, and subscriber ID (IMSI). We did not observe any applications use these data, and thus did not test any shadowing strategies for them.

EXFILTRATION BLOCKING: Exfiltration blocking to block exfiltration of data, we intercept calls to the network stack to (1) associate domain names with open sockets and (2) detect when tainted data is written to a socket. When an output buffer contains tainted data, we drop the buffer and choose one of two actions: we may drop the of- fending message covertly, misleading the application by indicating that the buffer has been sent, or overtly, emulating the OS behavior an application would encounter if the buffer were dropped as a result of the device entering airplane mode (all wireless connections disabled).

IV. THREATS IN ANDROID

There are several threats facing the android system and the following sections will list and explain some of the more common threats . A single malicious application can represent more than one of these threats TROJANS-generally speaking all android malware are Trojans because of the sandbox, the attack vectors used by viruses and worms are largely unavailable to the malware developers utilizing Trojans have thus become the norm. As with its desktop counterparts, the malicious code is usually included as a part of an otherwise legitimate looking application or added on to legitimate applications which are then redistributed as free applications on the third party markets .Applications misused for this purpose are often paid applications redistributed as free applications on the third party markets.

SPYWARE: Spyware are designed is malicious application designed to siphon off private information of one kind or another .Spyware are either commercial spyware or malicious. Commercial spyware are applications installed on the users handset manually to spy on the user while malicious spyware operates transmits data to a third party . ROOT EXPLOITS: Root exploit are created by legitimate members of the android community in order to gain control of their own devices rooting also gives the same amount of control to any application which gain access to the root rights

BOTNET: A botnet is a network of compromised devices usually computers which an attacker can use for his own purposes to transmit sensitive data as a part of the denial of service attack.

PREMIUM SMS SENDER: Some malicious applications are rather straight forward in their design where they ask for permission to send sms messages to premium rate numbers.

DRIVE BY DOWNLOAD: In drive by download the user is presented with a download pretending to be a system update when visiting a compromised website . if the user installs this false security update, the device is infected with a Trojan.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

PROOF OF CONCEPT: Proof of concept Trojans are usually the least dangerous and do not usually lead to large outbreaks .
These attacks are usually created for bragging rights or to demonstrate vulnerability.
DESTRUCTIVE TROJANS: Destructive Trojans aims to damage the infected devices or data stored on a device in some way or another .This can be via file corruption ,phone wiping or similar attacks
PHISHING: The android platform is vulnerable to phishing. Fake applications pretending to be real applications which when used would transmit the users login information to third party
CAPABILITY LEAKING: Applications leaking access to privileged device features providing other applications with access to features they should not have access to
INFORMATION LEAKING: Information leaks expose sensitive data to other applications on the device.

V. EXISTING SYSTEM

In mobile search the interactions between the user and search engines are limited by the small form factors of mobile devices. Also search engine does not think personally , it gives the results globally which are same for all users .And existing system not says about security of mobile if mobile is lost or stolen.

VI. PROPOSED SYSTEM

We propose a personalized mobile search engine, PMSE that captures the users' preferences in the form of concepts by mining their click through data. The user preferences are organized in an ontology-based, multi-facet user profile, and we are providing security for User profile if mobile lose or stolen there may be chance to hack user profile, for security purpose we can lock this mobile by sending lock command from server side application and if we will get back this mobile than we can restore it by using Wipe command, and we can use multi-facet user profile for adapt a personalized ranking function for rank adaptation of future search results. The technique used above is MAC based in which a

VII.MESSAGE AUTHENTICATION CODE (MAC)

In cryptography a message authentication code (MAC) is a short piece of information used to authenticate a message and to provide integrity and authenticity assurance affirm the messages origin . A MAC algorithm sometimes called a keyed cryptographic hash function is only one of the possible ways to generate MACs accepts as input a secret key and an arbitrary length message to be authenticated and outputs a MAC called tag . the mac value protects both a message data integrity as well as its authenticity by following verifiers who also possess a secret key to detect any changes to the message content.

.

VIII. RELATED WORK

Clickthrough data have been used in determining the users' preferences on their search results. Table 1, showing an example clickthrough data for the query "hotel," composes of the search results and the ones that the user clicked on (bolded search results in Table 1). As shown, cis are the content concepts and lis are the location concepts extracted from the corresponding results. Many existing personalized web search systems are based click- through data to determine users' preferences. Joachims [10] proposed to mine document preferences from clickthrough data. Later, Ng et al. [15] proposed to combine a spying technique together with a novel voting procedure to determine user preferences. More recently, Leung et al. [12] introduced an effective approach to predict users' conceptual preferences from clickthrough data for persona- lized query suggestions. Search queries can be classified as content (i.e., non-geo) or location (i.e., geo) queries. Examples of location queries are "hong kong hotels," "museums in london," and "virginia historical sites." In [9], Gan et al. developed a classifier to classify geo and non-geo queries. It was found that a significant number of queries were location queries focusing on location information. In order to handle the queries that focus on location information, a number of location-based search systems designed for location queries have been proposed. Yokoji [22] proposed a location-based search system for web documents. Location information was extracted from the web documents, which was

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

TABLE 1 Clickthrough for the Query "Hotel" converted into latitude-longitude pairs. When a user submits a query together with a latitude-longitude pair, the system creates a search circle centered at the specified latitude-longitude pair and retrieves documents containing location information within the search circle. Later on, Chen et al. [7] studied the problem of efficient query processing in location-based search systems. A query is assigned with a query footprint that specifies the geographical area of interest to the user. Several algorithms are employed to rank the search results as a combination of a textual and a geographic score. More recently, Li et al. [13] proposed a probabilistic topic-based framework for loca- tion-sensitive domain information retrieval. Instead of modeling locations in latitude-longitude pairs, the model assumes that users can be interested in a set of location- sensitive topics. It recognizes the geographical influence distributions of topics, and models it using probabilistic Gaussian Process classifiers. The differences between existing works and ours are. Most existing location-based search systems, such as [22], require users to manually define their location preferences (with latitude-longitude pairs or text form), or to manually prepare a set of location- sensitive topics. PMSE profiles both of the user's content and location preferences in the ontology- based user profiles, which are automatically learned from the clickthrough and GPS data without requir- ing extra efforts from the user. . We propose and implement a new and realistic design for PMSE. To train the user profiles quickly and efficiently, our design forwards user requests to the PMSE server to handle the training and rerank- ing processes. . Existing works on personalization do not address the issues of privacy preservation. PMSE addresses this issue by controlling the amount of information in the client's user profile being exposed to the PMSE server using two privacy parameters, which can control privacy smoothly, while maintaining good ranking quality.

MATHEMATICAL MODEL

Let Q = input is given as user query to PMSE server

Q = {HI,PS,NS}

Where,

HI is History which is maintained in History of search

PS is previous search information which maintain clickthrough data

NS is new search result which comes by using previous search + New Result

S is set of search engine

S = {s1,s2,s3,s4,……..,sn}

Identify the searching on servers

Q={q1,q2,q3,q4,……..qn}

Where Q is main set of searching query on server q1,q2,q3,q4,…..qn

q = L+C

L = set of location entropy

C = set of contents entropy

$$q(c) = -\sum_{i=1}^{k} P(c_i)\log P(c_i)$$

where k is the number of content concepts C={c1,c2,……,ck} extracted , $|c_i|$

$$|C| = |c_1| + |c_2| + |c_3| + |c_4| + \ldots\ldots + |c_k|$$

$$P(c_i) = |c_i| / |c|$$

$$q(l) = -\sum_{i=1}^{m} P(li)\log P(li)$$

m is the number of location concepts L={l1,l2,……,lm} extracted , |li| is the no of search results containing the location concepts li

|L| = |l1|+l2|+……+|lm|,

P(li) = |li|/|L|

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

Where
q={HI,CTD,NS,RSVM…..}
HI = History which is maintained in History of search
CTD=Maintain Click through Data on client side and send to RSVM
RSVM = Receive the Extracted Data on RSVM Server which are send through android mobile
NS = new search result which come by using previous search+New Result
SpyNB Method is the learns user behavior models from preferences extracted from click through data
SpyNB = {Po,U, →→PN}
Let Po be the positive set
U the unlabeled set, and
PN the predicted negative set
We get (PN (U) from SpyNB.
Po = {Po1,Po2,…….,Pon}
U = {u1,u2,…..un}
PN = {pn1,pn2,…..,pnn}
Identify the processes as P
P={set of processes}
P={P1,P2,P3,P4……..}

If(History found about CTD) then

P1 = {e1,e2,e3,e4}
Where
{e1=i|i is to search data selected search engine }
{e2=j|j is to retrieve information on search engine}
{e3=k|k is to send CTD to RSVM for reranking as user preferences}
{e4=l|l is to check GPRS Connection on android mobile}

If (No History found about the downloading of related data) then
P1={e1,e2,e4}
Where
{e1 = i|i is to search data on selected search engine}
{e2 =j|j is to retrieve information on search engine}
{e4=l|l is to check gprs connection on android mobile}
Overall subsets used in application
A={S,HI,P,PS,NS,Q}
Initial condition as Io
a) Android device should be activated the gprs connection
b) Admin have good internet connection

let A be the system that describes the mechanism of providing security to the user profile of a personalized mobile search engine by remotely locking and wiping it from a remote server using message authentication code technique
A={I,O,F,Su,Fa}
Where
I is the input to the system
O is the output to the system
F is set of functions
Su is the success of system
Fa is failure of the system

INPUT
I is the input set such that
I= {U,S,Q,M}

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 5, May 2015

U={u1,u2,u3,….,un} set of 'n' user profiles
S={s1,s2,s3,….sn}, set of 'n' search engines
Q={q1,q2…….,qn}
M={m1,m2,….,mn} set of 'n' user mobile registration profile

OUTPUT
O={E,L} is the output
E={e1,e2,…..,en} set of 'n' encrypted messages generated
L={L1,L2,……,LN} set of 'n' locked user profiles

FUNCTIONS
F is a set of functions where
F = {Fp,Fn,Fc}
Fp is the function of registering the user profile on the PMSE and producing the reranked result
Fn is the function of registering the mobile details of the user
Fc is the function of generating encrypted message
Tag : K*M → J is the tagging algorithm
Where
K is the secret key used to encrypt the message
M is the plaintext message (without the secret Key)
T is set of tags
J is the encrypted message
Vrfy:→ K*M→ J {YES, no}is the decryption algorithm
Success :
If Vrfyk (m, TagK(m)) = YES then t is a valid tag on message m
Failure :
If Vrfyk (m, TagK(m)) = NO , then t is not a valid tag on message m

IX. CONCLUSION

The system proposes the mechanism to provide security to the user by using remote lock/wipe which protects the users data and prevents the malicious user from launching DoS attack that sends such commands to the normal users intentionally.

REFERENCES

[1] android-apktool: Tool for reengineering Android apk files. http://code.google.com/p/android-apktool/.
[2] Privacy Blocker. http://privacytools.xeudoxus.com/.
[3] S. T. Amir Efrati and D. Searcey. Mobile-app makers face U.S. privacy investigation. http://online.wsj.com/article/ SB10001424052748703806304576242923804770968. html, Apr. 5, 2011.
[4] Apple Inc. iPhone and iPod touch: Understanding location services. http://support.apple.com/kb/HT1975, Oct. 22, 2010.
[5] A. R. Beresford, A. Rice, N. Skehin, and R. Sohan. MockDroid: Trading privacy for application functionality on smartphones. In Proceedings of the 12th Workshop on Mobile Computing Systems and Applications (HotMobile), 2011.
[6] J. Chow, B. Pfaff, T. Garfinkel, K. Christopher, and M. Rosenblum. Understanding data lifetime via whole system simulation. In USENIX Security Symposium, 2004.
[7] M. Egele, C. Kruegel, E. Kirda, and G. Vigna. PiOS: Detecting privacy leaks in iOS applications. In NDSS, 2011. [
8] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth. TaintDroid: An information-flow tracking system for realtime privacy monitoring on smartphones. In OSDI, 2010.
[9] W. Enck, M. Ongtang, and P. McDaniel. On lightweight mobile phone application certification. In CCS, 2009.
[10] A. Felt and D. Evans. Privacy protection for social networking APIs. In Proceedings of Web 2.0 Security And Privacy (W2SP), 2008.
[11] Google Inc. Android developers: Content providers. http://developer.android.com/guide/topics/ providers/content-providers.html.