Privacy Preservation of Mobile Data using Matrix Transformation

Sabarish BA
Department of Information Technology
Amrita School of Engineering
Amrita Nagar, Ettimadai, Coimbatore

Pradisa S
Department of Information Technology
Amrita School of Engineering
Amrita Nagar, Ettimadai, Coimbatore

Nithyasri J
Department of Information Technology
Amrita School of Engineering
Amrita Nagar, Ettimadai, Coimbatore

ABSTRACT
Data leakage from mobile phones and LBS is increasing with the exponential growth of technology, resulting in an innate risk of privacy threats. To address this issue, a novel method to protect user privacy is proposed in this paper. The proposed algorithms, namely SSET and SIMET, provide efficient means to encrypt the SIM and IMEI numbers respectively, thus protecting user identity. The reverse procedure to retrieve the numbers is provided by the algorithms SSDT and SIMDT and illustrated with examples. The main objective of this work is to secure the privacy of mobile phone users by employing optimal cryptographic techniques.

General Terms
Mobile Data Security, Algorithms, Privacy preserving.

Keywords
Privacy preserving, mobile data, security

1. INTRODUCTION
Advances in technology has led to tremendous increase in the usage of mobile devices for various applications. According to the International Telecommunication Union, there are 6.8 billion mobile subscriptions at the end of 2012, which is 97% of the world population [1]. The ubiquity of mobile devices has led to serious privacy concerns for its users.

Dissemination of information is the most demanded resource in today's globally networked society [2]. However, the release of microdata (specific information) provides access to the data provider's (mobile user) personal details, raising privacy concerns with respect to the embezzlement of induced information from this data [3]. The anonymity of the entities must be maintained in order to prevent leakage of confidential private information and the sensitive knowledge that can be mined from it.

1.1 Security vs. Privacy
The issues of privacy and security, though interrelated, are different. Data security deals with access control, authentication [4] and confidentiality of data while data privacy is defined as the appropriate use of data. Privacy is ensured and implemented by means of a security policy. While security provides means to avoid direct disclosure of data, it does not deal with inferences that can be drawn from the released data. In terms of RFID (Radio Frequency Identification), security can be defined as the ability of the system to protect the information transmitted between the tag and the reader from unauthorized users. Privacy is defined as the ability of the system to protect the meaning of the transmitted information from non-intended recipients.

1.2 Possible Attacks
Viruses, Trojans and other malware facilitate an attacker to access data in a user's mobile phone such as credit card number, bank details and could also render the services of the device inaccessible to its original owner. During an app installation, data could be sent to the app developer or a server without the knowledge of the customer [5]. Each app could access a particular detail and a number of such apps could correlate the data by using mobile number or IMEI (International Mobile Equipment Identity) number as a common factor [6].For example, consider an app that can access a person's location details (e.g., weatherbug) while another (e.g., cardiotrainer) can access name, age and gender details. It can be interpreted that a person X is currently at a particular location Y, thus making X vulnerable to privacy attacks [7].These specific details, when correlated with publicly available information such as name, driver's license number, etc. that can be collected from government offices (in the form of voters list, city directories) [2], enable the fabrication of the complete profile of a person. This kind of data grabbing is highly undesirable.

Downloading apps from trusted sources does not provide a complete solution as the data can be accessed by other means. LBS (Location Based Services) and GPS-enabled mobile phones provide customized services to users based on their location details. Moreover, many applications such as intelligent transport systems (ITS) track and collect vehicle locations to create traffic patterns [8]. But, an untrusted location server or a malicious user with access to these details could imperil an individual's privacy [9], leading to "location privacy". Moreover, LBS queries could reveal confidential information about an individual such as lifestyle habits, health conditions, religious affiliations etc. [10]. By tracking an individual's mobile phone usage pattern over a period of time, other sensitive information about personal and professional life is at a risk of exposure.

The method of extracting user's location through Geolocation system [11] involves identification of last-contacted and recently-contacted BS (Base Station) through SIM (Subscriber Identity Module) and IMEI numbers. Using SIM number, the IMEI number of the current mobile device onto which the SIM is embedded can be found. Thus, it becomes essential to

prevent a malicious user from accessing the SIM and IMEI numbers.

1.3 Motivation

The main motive of this work is to encrypt a user's sensitive data using optimal cryptographic techniques, thus making it indecipherable to an attacker. It is necessary to design a cryptographic function with minimum time and space complexity, in consideration of memory and processing power of the mobile devices.

Present day smartphones have impressive specifications such as 1.5 GHz quad core processors, gigabytes of storage capacity and multi-megabit connections [12]. However, it is essential to provide security to users of all types of mobile phones, including the generic devices that have specifications such as 100-200 MHz processor and 16MB of RAM. Therefore, the aim of this paper is to design a simple cryptographic technique that effectively provides security to mobile phone users and protects the data against attacks by using the minimum amount of resources in an efficient manner.

2. LITERATURE SURVEY

Information Disclosure relates to improper attribution of information to a mobile phone user. There are three types of information disclosures [13]. Identity Disclosure occurs when a user can be identified with the help of data obtained by the attacker. Attribute Disclosure occurs when the data reveals confidential information about the user. Inferential Disclosure occurs when the data makes it possible for the attacker to determine the value of some characteristic of the user with more accuracy than otherwise would have been possible. It is essential to prevent any type of information disclosure.

A method employed for protecting mobile data is cryptographic hash functions. A survey of Android apps revealed that 93% of the apps leak the IMEI numbers in the open while 7% attempt to provide security to the users by hashing the IMEI numbers [6]. While conversion of a string to its hash value is fairly simple, the reverse process of converting a hashed value to its original string is computationally difficult for the adversary unaware of the hash function. However, if the input strings are not random enough, then the hash function can be cracked with the help of lookup or rainbow tables. Besides, the inherent structure of the IMEI number drastically reduces the required number of computations/entries in a lookup table [6].

The first 8 digits of an IMEI number [14] represent TAC (Type Allocation Code) which depends on the model type of the mobile device. The next 6 digits uniquely identify the mobile device and are the most difficult to crack. [6] The last digit is the Luhn-checksum which is a function of the first 14 digits. Since the mobile market is predominantly dominated by a relatively small number of mobile device types, the adversary needs to build lookup tables for only selected TAC numbers. Hence, hashing mobile or IMEI numbers does not provide security. Employing salting technique, wherein random bits are added to the input of the hash function, fails if the adversary cracks the salt value. Thus, cryptographic hash functions fails to serve its purpose of protecting privacy.

The method of data masking leads to undesirable reduction of data quality [15]. Therefore, methods like sampling, swapping values, and adding noise disturb the data and do not satisfy "data quality" requirement. Another widely used method to protect user privacy is called k-anonymity which is implemented through generalization and suppression [2]. k-anonymity finds k users that are indistinguishable from each other such that an attacker cannot identify a single user out of the set. However, it employs the service of a TTP (trusted third party) to perform the anonymization. Consequently, this method fails if TTP is not trustworthy [16]. In addition, even if it is ensured that the release to each user satisfies k-anonymity, disclosure of sensitive data occurs if collusion happens or if a dataset is released to multiple users [17].

A 'quasi-identifier' is a set of attributes that uniquely identifies each respondent or tuple in a table. [18] On linking and co-relating with external sources of information, quasi-identifiers reduce uncertainty in the identification of a user. The quasi-identifier can also be called as 'leakage attribute' [18] because if it is cracked, the entire stored data can be retrieved. Various algorithms are available for identification of the quasi-identifiers such as Greedy minimal key algorithm, Random Sampling technique (used for large databases as samples are chosen at random) and computation of separation/distinct ratio (to identify minimal quasi-identifier) [18].In this project, it is assumed that the quasi identifiers have been identified to be the SIM and IMEI numbers of the mobile device. The two important methods for protecting content confidentiality are authentication and data encryption [19]. In this work, the method of data encryption is employed to provide privacy to users.

3. PROPOSED ALGORITHMS

The following algorithms are proposed for efficient encryption of SIM and IMEI numbers. The decryption procedure for retrieval of the encrypted numbers has also been explained and illustrated with examples. The computational complexity of the algorithms are kept to a minimum without compromising on the level of the security that is provided for the users. This ensures that on implementation in a mobile phone, the algorithms do not overload the processor.

3.1 Algorithm SSET (Secure SIM Encryption Transformation)

1. Calculate the total sum (TS) of the digits in the SIM number.
2. Divide the number into pairs { m_x , m_{x+1}} for each j where m_x and m_{x+1} are digits in the Partition j, where j = 1,2,3,4,5. Calculate the Sum of each partition S_j where j = 1, 2, 3, 4, 5.
3. Calculate Key value for each partition

   for j->1 to 5

   { K_j = TS % S_j }
4. for j-> 1 to 5

   { E = { Union of all E_j where E_j is result set of circular shift operation on each digit in Partition j by K_j }}
5. Encrypted set is E = {E_1,K_1 ; E_2,K_2 ; … ; E_5,K_5}

3.1.1 Example

Let the SIM number be 8783653721

1. TS=15+11+11+10+3=50
2. {m_1, m_2}= { 8, 7}

   {m_3, m_4}= {8, 3}

   {m_5, m_6}= {6, 5}

   {m_7, m_8}= {3, 7}

   {m_9, m_{10}}= {2, 1}

S_1=15, S_2=11, S_3=11, S_4=10, S_5=3

3. K_1=50%15=5

   K_2=50%11=6

   K_3=50%11=6

   K_4=50%10=0

   K_5=50%3 =2

4. E_1= Circular shift of m_1 and m_2 by K1 times (m_1=8, m_2=7, K_1=5)

   E_1=32

   Similarly,

   E_2=49

   E_3=21

   E_4=37

   E_5=43

5. Hence, the encrypted mobile number along with the key is- E={32,5 ; 49,6 ; 21,6 ; 37,0 ; 43,2 }

3.2 Algorithm SSDT (Secure SIM Decryption Transformation)

Given set: {E_1,K_1 ; E_2,K_2 ; … ; E_5,K_5}

1. for j-> 1 to 5

{Perform a reverse cyclic shift operation on each digit of Ej by K_j = { m_x , m_{x+1}}}

2. Group the set of { m_x , m_{x+1}} values obtained for the Partition j to form a number where j->1 to 5.

3.2.1 Example

1. E_1=32 and K_1 =5

Reverse cyclic shift operation of each digit in E_1 yields: 8 and 7.Thus for E_1 corresponding m_1 and m_2 are 8 and 7.

2. Similarly, the other values are obtained and

grouped together as follows, M= {87, 83, 65, 37, 21}

3.3 Algorithm SIMET (Secure IMEI Matrix Encryption Transformation)

1. Partition the number into groups Gj with 3 digit each where j-> 1 to 5 and a=5 i.e the total no of partitions.

   for j-> 1 to 5

   {Swap (G_j, Ga_{-j+1});}

   Merge all G_j where j -> 1 to 5

2. Calculate the total sum (TS) of the digits in the IMEI.
3. Divide the number into three partitions P_j having five digits {m_x, m_{x+1},m_{x+2},m_{x+3},m_{x+4}}

where m_j and m_{j+1} are the digits in the Partition j, where j = 1,2,3 Calculate the Sum of each partition Sj where j = 1,2,3.

4. Calculate Key value for each P_j

   for j->1 to 3

   { K_j = TS % S_j}

5. for j-> 1 to 5

{E = { Union of all Ej where Ej is result set of circular shift operation on each digit in Partition j by Kj }}.

6. Let M_a be the Matrix representation of the above values, each partition is represented by as a row in the matrix.

M_a =

m_1	m_2	m_3	m_4	m_5	K_1
m_6	m_7	m_8	m_9	m_{10}	K_2
m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	K_3

Split the above 6x3 matrix into two 3x3 matrices A and B.

A=

m_1	m_2	m_3
m_6	m_7	m_8
m_{11}	m_{12}	m_{13}

B=

m_4	m_5	K_1
m_9	m_{10}	K_2
m_{14}	m_{15}	K_3

7. Calculate and transmit the inverses of the matrices A and B

3.3.1 Example

Let the IMEI number be 355293046169243

1. 355 293 046 169 243

   Performing swap operation,

   243 169 046 293 355

2. TS = 2 + 4+ 3 + 1 +...+ 5 = 62
3. P_1 = 24316; P_2=90462; P_3=93355;

   S_1=16; S_2=21; S_3=25;

4. K_1= 62 % 16 = 14;

   K_2 = 62 % 21 = 20;

   K_3 = 62 % 25 = 12;

5. E_1 = P_1 Shifted K_1 times = 68750;

E_2 = 90462 shifted 20 times = 90462;

E_3 = 93355 shifted 12 times = 1557;

6. M_a=

6	8	7	5	0	14
9	0	4	6	2	20
1	5	5	7	7	12

A=

6	8	7
9	0	4
1	5	5

B=

5	0	14
6	2	20
7	7	12

7. A^{-1}=

0.15	0.03	-0.24
0.30	-0.17	-0.29
-0.33	0.16	0.54

B^{-1}=

0.61	-0.52	0.14
-0.36	0.20	0.08
-0.14	0.18	-0.05

3.4 Algorithm SIMDT (Secure IMEI Matrix Decryption Transformation)

1. From the given two matrices say C and D, find the respective inverses which yields the matrices A and B.

A=

m_1	m_2	m_3
m_6	m_7	m_8
m_{11}	m_{12}	m_{13}

B=

m_4	m_5	K_1
m_9	m_{10}	K_2
m_{14}	m_{15}	K_3

2. Combine the two matrices to form a new matrix M_a.
M_a=

m_1	m_2	m_3	m_4	m_5	K_1
m_6	m_7	m_8	m_9	m_{10}	K_2
m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	K_3

3. Each row in the matrix Ma represents a partition j, where j= 1 to 3.

for j-> 1 to 3

{Perform reverse shift operation on each element m_x of the row using K_j }

4. Combine the obtained result from step 3 and partition the

Number into groups G_j with 3 digits each where j= 1 to 5 and a=5 where a represents the total number of partitions.

for j->1 to 5

{ Swap (Gj, Ga-j+1); }

Merge all Gj to form G where j->1 to 5.G is the decrypted IMEI number.

3.4.1 Example

1. Let the given matrices be
C^{-1}=

0.15	0.03	-0.24
0.30	-0.17	-0.29
-0.33	0.16	0.54

D^{-1}=

0.61	-0.52	0.14
-0.36	0.20	0.08
-0.14	0.18	-0.05

Calculating the inverses of the above matrices, we get C and D,

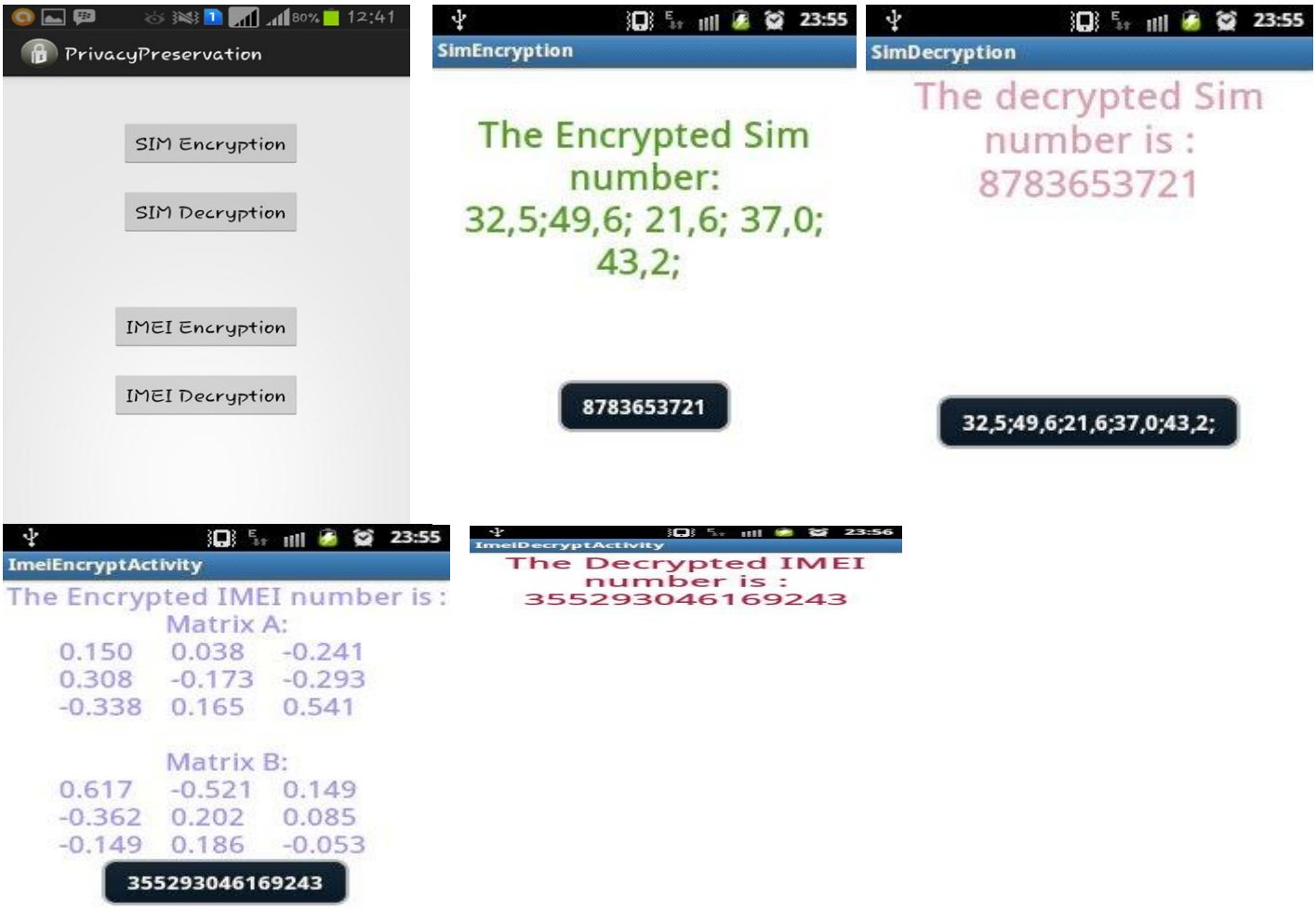

Fig 1: Implementation of the algorithms in android

C=

6.62	7.9	7.2
9.6	-0.2	4.15
1.18	4.94	5.03

D=

5.02	0.16	14.3
6.01	2.24	20.4
7.58	7.62	12.3

2. Combining C and D to form M_a,

M_a=

6	8	7	5	0	14
9	0	4	6	2	20
1	5	5	7	7	12

3. From M_a, we get,

{ 68750,14 ; 90462,20; 15577,12 }

P_1 = 68750 right shift 14 times = 24316;

Similarly,

P_2 = 90462;

P_3 = 93355;

4. Merging the partitions,

24316 90462 93355

Dividing into G_j,

243 169 046 293 355

Swapping,

355 293 046 169 243

Merging,

G=355293046169243

where G is the decrypted IMEI number.

Table 1. Execution Time of the Algorithms

Algorithm	Execution time(microseconds)
SSET	5 - 15
SSDT	3 - 5
SIMET	19 – 22
SIMDT	20

4. CONCLUSION

In this paper, a new approach was proposed to combat the issue of data leakage from mobile phones, leading to privacy concerns. The proposed set of algorithms provide a layer of security to mobile phone users with minimal computational complexity, thus making it employable for security in both smart phones as well as older models of mobile phones. The techniques can also be modified and extended for use in relation to other algorithms and encryption techniques, thereby increasing the computational complexity and the security factor.

The future course of action for the project involves the integration of the proposed algorithms with apps that provide data-sharing services and apps that provide location-based services. Encrypting the SIM and IMEI numbers, in this case, will prevent the disclosure of the individual's identity.

5. ACKNOWLEDGMENTS

We would like to extend our sincere thanks to our family and friends for helping and motivating us during the course of the project. We would like to thank all those who have helped, guided and encouraged us directly or indirectly during the project work. Last but not the least, we thank God Almighty for the blessings which made our project a success.

6. REFERENCES

[1] Favell, Andy. 2013. Global mobile statistics 2013 Part A: Mobile subscribers; handset market share; mobile operators. http://mobithinking.com/mobile/mobile-marketing-tools/latest-mobile-stats/a.

[2] Ciriani, V., Vimercati, De Capitani di., Foresti, S., and Samarati, P. K-Anonymity. 2007. Advances in Information Security. Springer US.

[3] Gkoulalas-Divanis, Aris and Verykios, Vassilios S. 2009. Exact Knowledge Hiding through Database Extension. IEEE Transactions on Knowledge and Data Engineering, Vol. 21, No. 5.

[4] Sweeney, Latanya. 2002. K-Anonymity: A Model for protecting Privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems.10 (5): 557-570.

[5] Tung, Liam. 2013. Google Play privacy slip-up sends app buyers' personal details to developers. http://www.zdnet.com/google-play-privacy-slip-up-sends-app-buyers-personal-details-to-developers-7000011249/

[6] Gagnon, Michael N. 2011. Hashing IMEI numbers does not protect privacy.http://blog.dasient.com/2011/07/hashing-imei-numbers-does-not-protect.html

[7] Angwin, Julia. 2012. What They Know- Mobile series, The Wall Street Journal. http://blogs.wsj.com/wtk-mobile/

[8] Gidofalvi, Gyozo., Huang, Xuegang and Pedersen, Torben Bach. 2007. Privacy-Preserving Data Mining on Moving Object Trajectories. IEEE.

[9] Dong, Changyu and Dulay, Naranker. 2011. Longitude: a Privacy- preserving Location Sharing Protocol for Mobile Applications. IFIP Advances in Information and Communication Technology.

[10] Khoshgozaran, Ali., Shahabi, Cyrus and Shirani-Mehr, Houtan. 2010. Location privacy: going beyond K-anonymity, cloaking and anonymizers. Springer-Verlag London.

[11] Rahul. 2009. Tracking and Positioning of mobile devices in Telecommunication networks. http://www.slideshare.net/rahul_2013/tracking-and-positioning-of-mobile-systems-in-telecommunication-networks-15633468

[12] Fitchard, Kevin. 2012. The super-computing phone: At&T's predictions for devices in 2020. http:://gigaom.com/2012/09/10/the-super-computing-phone-atts-predictions-for-devices-in-2020/

[13] Samarati, Pierangela. 2008. K-anonymity. Foundations of Security Analysis and Design (FOSAD).

[14] Admin. 2010. IMEI Structure. IMEI Tools- Analysis tools, manuals, instructions. http://imei-number.com/imei-structure/

[15] Vijayarani, S., and Tamilarasi, A. 2010. K-Anonymity Techniques - A Review. International Journal of Computer Science and Application.

[16] Wernke, Marius., Skyortsov, Pavel., et al. 2012.A classification of location privacy attacks and Approaches. Springer.

[17] Pei, Jian., Tao, Yufei., et al . 2009. Privacy Preserving Publishing on Multiple Quasi-Identifiers. Proceedings of the Twenty-fifth IEEE International Conference on Data Engineering (ICDE'09).

[18] Motwani, Rajeev and Xu, Ying. 2008. Efficient Algorithms for Masking and Finding Quasi-Identifiers.

[19] Pai, Sameer., Bermudez, Sergio., et al. 2008. Transactional Confidentiality in sensor Networks. IEEE Security and Privacy.