SmartSec: Secret Sharing-based Location-aware Privacy Enhancement in Smart Devices

Bett Ben Chirchir, Xiaokuan Zhang, Mengyuan Li, Qiyang Qian, Na Ruan, Haojin Zhu
Shanghai Jiao Tong University, Shanghai, China
{nebroris, number_x, limengyuan, whiteqqy, naruan, zhu-hj}@sjtu.edu.cn

***Abstract*—In the recent past we have witnessed a quick rise of smart devices, including smart phones, tablets, smart watch, smart television as well as other smart devices. The smart devices store quite a lot of sensitive personal data (e.g., private photos). These devices can breach one's privacy if information in the devices are accessed by unauthorized users. Therefore, smart devices are facing threats of privacy leaking and how to protect privacy of smart devices is regarded as an important research challenge. In this paper, we propose a system, coined as SmartSec, which employs secret sharing techniques among multiple smart devices to enhance privacy protection. In particular, SmartSec can achieve privacy enhancement by storing sensitive data (e.g., photos) on at least three devices and requiring at least two of them to access the original data. This will thus secure the data from any unauthorized attackers because any leakage of one piece of secret is meaningless. Further, secret sharing will introduce extra computation and transmission overhead. To reduce such an overhead and optimize the performance, another important feature of SmartSec is that it could recognize and classify different locations by exploiting location tags and then apply different security policy to different locations to achieve the tradeoff of security and performance. For example, a user can demarcate safe regions (e.g., home area) and the sensitive data will be automatically recovered in home region, distributed and stored/masked at different devices in public regions, where the process is automated which provides the user a transparent experience. We implement the system and conducted a series of experiments on the prototype to evaluate the results.**

Keywords – **Smart Devices, Location Tag, Secret Sharing**

I. INTRODUCTION

In the current digital age, a smartphone is one of the most important and depended on gadget by human beings to run our day to day activities. With bigger capacity, a smartphone can carry huge amounts of data from the most basic like phone call logs, text messages to most private and sensitive data like photos or even passwords, holding such data in our hands we risk exposing it to the public for example data leakage when connected to public networks. The sky rocketing use of smart devices means increasing threat to privacy if they are accessed by strangers and compromised. These then brings forth the question: How can we protect our privacy at the same time embrace technology? Privacy has attracted attention to many through cyber crimes, In 2014, Apple's cloud, icloud was attacked and celebrity photos were leaked [1]. As much as people are talking about it in low tones, it needs to be addressed.

Android has the largest share in smartphones operating system market to date, according to International Data Corporation (IDC) report [2]. In 2014, global mobile phone shipments reached 1.3 billion and Android phones accounted for more than one billion and owned 81.5% of the total market share. Therefore most of the current research is focusing on Android platform.

Even though smartphones hold more sensitive data, the majority of academic study of Android system is revolving around vulnerabilities but little attention is paid to privacy protection issues on the Android system. Many applications request permission to read contacts, text messages and other privileges,users can easily be led to disclosure of some sensitive information. Take for example, when you install an application requesting permission to read your address book so that it can recommend you, users who use the same application. In addition, it can also request permission to read your photos so that you can be able to share photos and friends too. The application will be able to read without your knowledge your address book and phone's photos hence your privacy will be exposed. Other applications can access your information without permissions as studied by Grace in [3].

Android is dominant in the smartphone market and open source. We therefore propose SmartSec, an Android system for privacy protection. SmartSec enhances privacy of sensitive photos in smart devices through secret sharing technology. A photo is an essential part of our lives and carries a lot of personal information thus its protection is critical. If one loses a smart device the concern the person will have is of the most sensitive information. The proposed system addresses privacy of sensitive photos in smartphones, which is one of the greatest challenge human beings face, when a device is lost or attacked while enabling a user to freely interact with protected data.

Secret sharing, leveraged by SmartSec is basically dividing a secret into many pieces and storing all the pieces separately. One piece cannot reveal any information about the secret. To reveal the secret, a number of pieces need to be assembled together (The number is defined by user). Over the years, secret sharing has played a major role in security and in this paper we introduce it to smart devices, the goal is to protect photos from leaking information when devices containing the information are attacked. First, we select the sensitive part of the photo where we want to protect. Then we share this sensitive part into several pieces. Each piece contains some information of the secret. The most fascinating part of secret sharing is that after sharing, one piece cannot reveal any

978-1-5090-0243-6/15/$31.00 ©2015 IEEE

information whatsoever of the secret. To reveal the secret a minimum number of shares must be assembled together. In SmartSec, we share information to three devices and require two to reveal the secret. When these devices are compromised, they will not leak any information because the information to be meaningful it needs the other parts.

In addition to secret sharing, SmartSec features a novel technology, location tags. SmartSec leverages locations tags in order to reduce extra computation and transmission overhead. Location tags allows a user to define some region (e.g Home area) where the user's data is safe. In this region, user's data will be automatically pulled from other devices and reconstructed. When the user is out of the safe region, the data will be automatically secured by secret sharing to other devices.

Our main goal is to protect privacy of user's sensitive information while making it available. SmartSec achieves this goal by employing secret sharing technology.

The paper is organized as follows: Section II discusses related work, system design is presented in Section III, we evaluate conducted experiments in Section IV and conclude in Section V.

II. RELATED WORK

Secret sharing schemes have received some breakthroughs since its introduction in 1979 by Shamir [4]. With a lot of applications in technology, it has encountered a number of challenges as well. Starting from protecting keys for encryption where a key was divided into several shares and shared to several people, it faced the challenge of untrusted personnel who collaborated to form the key.

Secret sharing is mostly applied in applications and developed images which introduced Visual Secret Sharing (VSS). VSS was first introduced in 1995 by Shamir [5]. They divided a photo into several shares and superimposed them together to form the original image. The major drawback in this development was increased size of the reconstructed image. Chen and Chang in [6] introduced an efficient multi secret image sharing which addressed some of these drawbacks.

Askari in [7] proposed new schemes that solved the capacity issue and lack of clarity in the reconstructed images. This drew a lot of attention. Chen and Chan in [8] introduced hierarchical threshold secret image sharing. All these was to solve distortion in the reconstructed photo. Wu in [9] brought forward a user friendly secret sharing scheme with reversible steganography based on cellular automata that achieved lossless reconstruction.

In SPREAD [10], Lou et al. employed secret sharing to enhance data confidentiality in mobile adhoc networks where data was shared and relayed through different paths to the destination. Any compromised share is of no use since it is dependant on the others to reveal information.

A. SECRET SHARING

Secret sharing, also known as secret splitting, refers to a method to share a secret among some people and each of these people will hold a secret part called a share. They are also called secret holders. The secret can be restored if and only if a sufficient number of secret share holders are combined. One holder with a share is not able to view the secret thus one share is meaningless. Secret sharing was first introduced by Adi Shamir [4] and George Blakley [11] in 1979. Adi Shamir proved that if you divide data D into n pieces, D can be easily reconstructed from k pieces but complete knowledge of $k-1$ pieces reveals absolutely no information about D. This scheme is known as (k, n) threshold scheme. In this paper, we apply the proposed Adi Shamir (k, n) threshold scheme, (t, n) threshold scheme where t is the minimum number of shares that if combined will reveal the secret.

B. LOCATION TAG

A location tag is a secret associated with a point in space and time. An efficient location tag should have two main features:

$1) Reproducibility$: If two measurements at the same place and time yield t_1 and t_2, then t_1 and t_2 should match with high probability.

$2) Unpredictability$: An adversary not at a specific place or time shouldn't be able to produce a tag that matches the measured tag at that location at that time.

Location tag has been applied in [12] to check if two users are close to each other without revealing their personal information. Location cheating has been studied where they proof that a user is at a location that he claims he is [13]. Boneh succeeded in generating a tag from a noisy environment by the use of fuzzy extractor which can take radio frequency signals as inputs like WiFi, cellular signals, TV as well as non radio frequency signals like infrared and ultra sound [14]. This therefore widens the scope of tag generation. In this paper, we discuss a few location tag generation methods:

1) Global System for Mobile Communication (GSM): The network of base stations which communicate oftenly by sending and receiving signals from users can be a good source for a location tag. Since every base station has a unique identification, the control messages sent between the base station and its terminals can be used to generate a location tag. Temporary mobile subscriber identity (TMSI) which is unique to every terminal can be extracted from the messages as implemented in [15] using $4G$ network.

2) Bluetooth: Every device with bluetooth has unique bluetooth identification which can be a source though it has a limited serving range.

3) Environmental Noise: This is unique at a given time and place therefore it is impossible to regenerate hence can be a viable source.

4) WiFi: WiFi traffic as well as access point IDs can be a good source. Broadcast packets taped, studied and compared to other networks at the same time will be unique in different networks. In [12], packets were collected from two different devices and after comparison, 90% of the packets were common to both devices. The drawback in WiFi is that a network can be wide thus it will be difficult to define proximity.

5) Global Positioning System(GPS): GPS is a satellite base system made up of a network of orbital satellites which works by timings sent by a number of orbital satellites. Studies by Sherman [16] shows that signals from four or more satellites forms a secure location tag is unpredictable.

III. SYSTEM DESIGN

Our system consists of but not limited to three devices (smart phone, watch, glasses) and cloud. Our main goal is to protect privacy of sensitive data by secret sharing scheme and be able to rebuild it on certain conditions. The user captures a photo with a specified camera and selects the sensitive part of the photo.Then secret sharing is applied to the sensitive part after which it is sent to other devices and a backup of the insensitive part to the cloud. During reconstructing the user must have at least two of the three devices to rebuild the photo. The overview of the system is shown in Fig. 1.

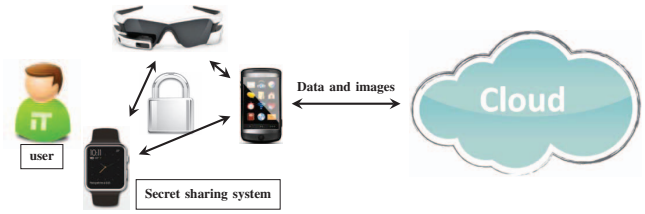


Fig. 1. System Overview

To further optimize the design, we added an automated process of sharing and recreating the photo by declaring some safe regions where the user doesn't need privacy protection.This is implemented through the use of location tags.

A. SYSTEM ARCHITECTURE

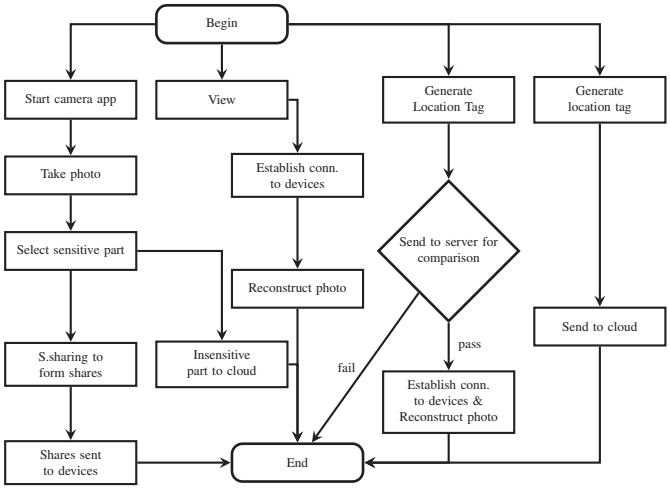


Fig. 2. System Flow Outline

At first, the user selects a specific camera, captures a photo, then selects the sensitive part of the photo and submits it for protection. In the protection phase, the phone will first establish a connection to other devices through Bluetooth connection, then send the shares to them. The sensitive part of the photo will be safe as it will be split in three secret shares. Each share can't reveal any information unless two of the three devices are brought together. The phone will store the insensitive part and also send it to the cloud. The outline is shown in Fig. 2.

The user can view the photo anytime as long as there are at least two devices nearby. The user will then establish a connection to the available device and be able to reconstruct the photo. Fig. 2 outlines the process for viewing the photo.

The outline shown in Fig. 2 is the original scheme from which we optimize it by the use of location tags. Fig. 2 shows the steps involved. The user needs to satisfy the condition of being in a safe location to view the photo, then the original scheme runs automatically; Connect to any two devices then view the photo. To generate a location tag the user needs a connection to the three devices as shown in the far right of Fig. 2.

B. SECRET SHARING

In order to protect the user's private data, we have developed an Android system based on secret sharing, SmartSec. This system protects smartphone user's private photos. Users can opt to use this system's camera when capturing photos with sensitive or private information. First, the system will automatically search and connect to the other playing devices (owned by the same user since it will store information for later processing) and will prompt the user to connect. Secondly, it will divide the captured photo into 16 pieces which then allows the user to select the sensitive part. Lastly, it will apply secret sharing scheme to the selected part of x blocks $(1 \leq x \leq 16)$ then send the parts to other devices. The phone will send a backup of its share to the cloud for redundancy. The backup means nothing to the cloud and to any attacker, since the information is unreadable without the other shares and therefore cannot read the backed up share from the smartphone. The system will be able to protect the user's private photo by employing secret sharing scheme.

Employing (k, n) threshold scheme by Adi Shamir [4] , the goal is to convert a secret S into n shares $S_1, S_2, ..., S_n$ respectively which satisfies the following two conditions:

(1) S can be easily calculated with any k or more shares of S_i.

(2) $k-1$ or less shares of S_i learns completely nothing about S.

Suppose we want to use the (k, n) threshold scheme to share a secret S, without loss of generality, we assume that it is an element of the finite field GF (P), which satisfies $0 < k \leq n < P$; $S < P$; P is a prime number. We randomly select $k-1$ positive integers $a_1, ..., a_{k-1}$, which satisfies $a_i < P$, and let $a_0 = S$ and construct a polynomial:

$$f(x) = (a_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... + a_{k-1} x^{k-1}) mod P \quad (1)$$

We can design any configuration of n points from equation (1). For example, configuration $i = 1, ..., n$ to obtain $(i, f(i))$. Thus, each of the secret holders can share a point

from every person's share. If we get this n, any k points, we can obtain an original polynomial $f(x)$ using Lagrange polynomial. Secret S is the value of the coefficient a_0. The calculation method is as follows: Set up k points coordinates $(x_1, y_1), (x_2, y_2), ..., (x_k, y_k)$. For jth polynomial $l_j(x)$, such that

$$l_j(x) = \prod_{i=1, i\neq j}^{k} \frac{x - x_i}{x_j - x_i} \qquad i, j = 1, 2, 3, ..., k \tag{2}$$

The original polynomial:

$$f(x) = \sum_{j-1}^{k} y_j l_j(x) \tag{3}$$

In our scenario, $k = 2$, $n = 3$, that is a total of three smart devices, from which any two can restore the original photo. Since our program is designed for image therefore we need to adjust the original formula. The display uses RGB color standard where it varies the Red (R), Green (G) and Blue (B) strength and overlaps them to obtain a variety of colors. The image is constituted by a plurality of pixels, and each pixel's information (r, g, b) is within the range of $0 \leq r, g, b \leq 255$. The largest prime number in $0 - 255$ is 251, so pixels with values greater than 250 are truncated to 250. This result in little distortion in the restored photo. However, pictures with RGB values greater than 250 are not much, so the amount of data loss is minimal. After getting the RGB values of the pixels from the picture, we apply secret sharing scheme transformation to hide the details. From our scenario, $k = 2$, $n = 3$, thus we propose the following secret sharing equation:

$$f(x) = a_0 + a_1 x \tag{4}$$

From equation (2) a_0 is an RGB value, a_1 is a random number within the range 0 to 249. In order to facilitate the calculation, we use the same a_1 in the same point, and set x to 1,2,3. For example, we have original pixel values (r, g, b). After applying secret sharing we get $(r+a_1, g+a_1, b+a_1), (r+2a_1, g+2a_1, b+2a_1), (r+3a_1, g+3a_1, b+3a_1)$ respectively. The obtained results are 3 meaningless parts of pictures which are the shares that are sent to the other devices.

C. RECONSTRUCTION

From Equation (2) to (4), we were able to get three shares from which two of any of them can restore the original image pixel by pixel. Since $k = 2$, $n = 3$, we assumed the two pixels were (r_1, g_1, b_1) and (r_2, g_2, b_2). Exemplifying r_1 and r_2, we set $y_1 = r_1 = a_0 + a_1 x_1$, $y_2 = r_2 = a_0 + a_1 x_2$. By Equation (2), we can calculate

$$l_1(x) = \frac{x - x_2}{x_1 - x_2}, \qquad l_2(x) = \frac{x - x_1}{x_2 - x_1} \tag{5}$$

By Equation (3) we obtain:

$$f(x) = \sum_{j=1}^{2} y_j l_j(x) = \frac{y_1 - y_2}{x_1 - x_2} x + \frac{y_2 x_1 - y_1 x_2}{x_1 - x_2} \tag{6}$$

By equation (6), we can observe that, a_0 is $f(x)$, the constant term, namely:

$$a_0 = \frac{y_2 x_1 - y_1 x_2}{x_1 - x_2} \tag{7}$$

According to the formula (7), we can calculate the image in RGB values for each pixel, and thus reconstruct the picture. It is completely consistent with the original image. Fig. 3 shows the structure in detail.

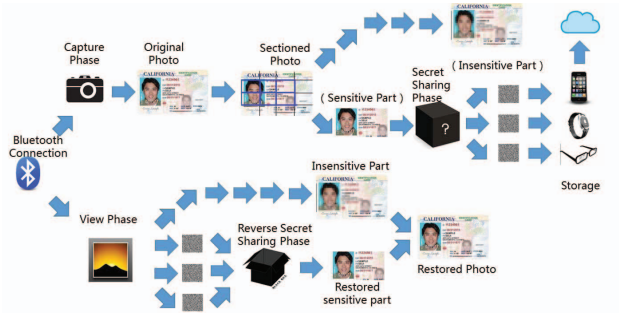


Fig. 3. Complete Structure

D. LOCATION TAG

To facilitate our experiment, we implement our system using GPS coordinates. We develop a tag from the coordinates by generating a 36 bit code. We take total longitude/latitude degrees divided by two and determine if our phone's longitude/latitude falls on the left or right side of the half, if it is on the left then our tag is 0 and if right 1 then we continue by appending the rest on them until a certain threshold which is determined by the area we would like to be our safe region. After that we append both the longitude and latitude together to get our location tag. If a device is in the safe region the tag will be the same. The tag doesn't have any information of the location hence preserves the location of the user.

IV. EXPERIMENTAL EVALUATION

We demonstrate our experiment in two parts: 1) We discuss the secret sharing process and show the results of the system. 2) We present optimization of the original scheme.

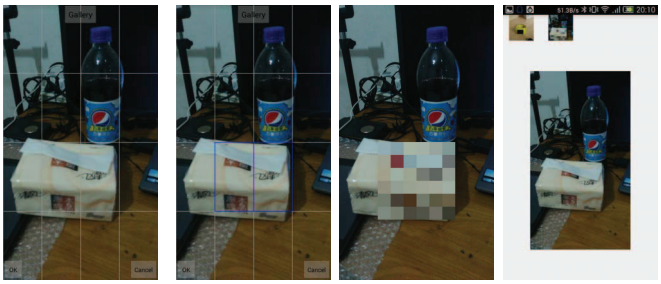
A. ORIGINAL SCHEME

In this part we demonstrate connection of devices, sharing of photos by secret sharing and reconstruction of the photo. Our experiments were conducted based on three devices: Host: Nubia $Z7$ mini running Android 4.4; client: tablet Nokia $N1$ running Android 5.0, and Samsung galaxy note $N7000$ running Android 4.1.

1) connection: We connect our three devices through Bluetooth connection. The devices must be connected before the sharing process.

2) sharing process: First, a photo is taken with the application camera. Fig. 4(a) shows a photo after it has been captured. The photo will be divided into 16 sections/frames after confirmation for privacy protection. The user then selects the sensitive part from the photo and the selected frames will turn blue as shown in Fig. 4(b). The selected part will then be shared and sent to other devices. Fig. 4(c) shows the photo after protection.

3) reconstruction: After pairing to one of the devices, the photo can be reconstructed. The conditions for reconstruction is connect to one of the devices and can be called by the user or when the user is in a safe region. Fig. 4(d) shows the reconstructed photo.



(a) Cap Pic (b) Selected Pic (c) Prot. Pic (d) Orig. Pic

Fig. 4. Sharing and Reconstruction Photo Process

B. LOCATION-TAG OPTIMIZATION

It is accomplished through the use of location tag which is first generated for the safe region and stored in the cloud. Our location tag generation algorithm is shown in algorithm 1. The location tag defines a specific region and the one used is a section of 100 squared meters thus the tag changes after every $100m$.

1) Dataset: Fig. 5 shows our sample dataset that was collected by a HTC phone.The data, collected on campus by strolling with the phone for around 2hrs and a ride by local metro covering 11.5km and back. Half of the dataset was by metro and the other inside campus.

The first column shows our location tag, the next is the index which is derived from the location tag. Same numbers mean similar tags or tags within the same region. Different numbers mean tags from different regions. The third column shows the distance between two tags next to each other. The last two columns are our gps coordinates collected by the smartphone, latitude and longitude respectively.

LOCATION TAG	INDEX	DISTANCE BETWEEN TAGS	LATITUDE	LONGITUDE
10101100000111101011010110001011001 00	0	0.0000	31.02131515	121.4273347
10101100000111101011010110001011001 01	1	109.2471	31.02156418	121.4284424
10101100000111101011010110001011001 01	1	9.7370	31.02159207	121.4285392
10101100000111101011010110001011001 01	1	12.3298	31.02169157	121.428596
10101100000111101011010110001011001 01	1	12.9368	31.02180723	121.4285827
10101100000111101011010110001011001 01	1	9.0064	31.02186618	121.4285181
10101100000111101111010110001011001 01	2	14.7654	31.02198578	121.4284512
10101100000111101111010110001011001 01	2	10.2654	31.02203716	121.4283618

Fig. 5. Dataset Sample

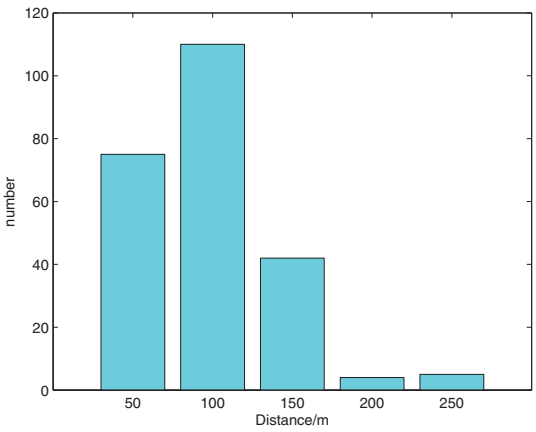
Algorithm 1: Location Tag Generation

```
while Number of bits < 18 do
  if Latitude > CenterValue then
    LeftValue = CenterValue
    CenterValue = LeftValue + RightValue / 2
    Tag.append(1)
  end if
  if Latitude < CenterValue then
    RightValue = CenterValue
    CenterValue = LeftValue + RightValue /2
    Tag.append(0)
  end if
end while
while Number of bits < 18 do
  if Longitude > CenterValue then
    LeftValue = CenterValue
    CenterValue = LeftValue + RightValue / 2
    Tag1.append(1)
  end if
  if Longitude < CenterValue then
    RightValue = CenterValue
    CenterValue = LeftValue + RightValue /2
    Tag1.append(0)
  end if
end while
Tag.append(Tag1)
```

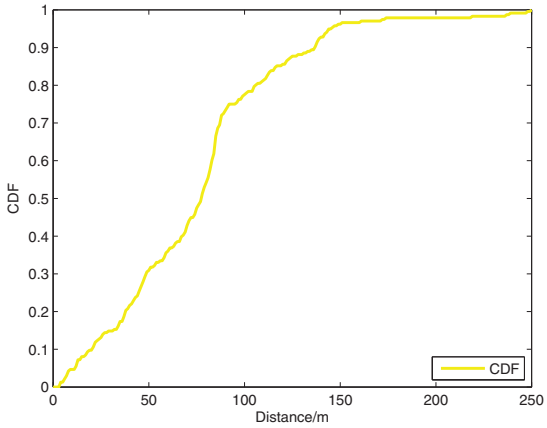
2) Location-Tag Analysis: This system generates a new tag for comparison every $10m$ and $1m$ (half of our dataset applied $10m$ and the other $1m$) although it can be configured to any figure according to the user's needs.Then the comparison between the generated tag and the one stored in the cloud follows and if they match, the photo will be automatically reconstructed. Fig. 6a shows a bar graph of location tags and distance between each tag. Majority of the tags lie at $100m$ apart since we defined an area of 100 squared meters. The numbers represent location tags, a number can represent more than one tag.The average distance between tags is 5.9706 from the configured $10m$ and $1m$ from a data set of 3155 tags spanning a total distance of about $28km$ therefore our method of tag generation is feasible. Fig. 6b shows cumulative distribution function of the location tags. It shows that around 80% of the tags cover $100m$, which means our method achieves high precision.

V. CONCLUSION

In this paper, we proposed SmartSec, a system that enhances privacy of a user's sensitive photo. Our main goal is to prevent sensitive photo leakage once user's data is attacked. The proposed scheme is based on secret sharing. The scheme shares sensitive parts of a photo to multiple devices and requires at least two devices to view the photo hence protects user's data. In addition, the scheme incorporates location tags to optimize the process making it location aware. Other applications of



(a) Graph of Tags and Distance



(b) CDF

Fig. 6. Data Analysis

this scheme in smartphones and other devices can be explored in future.

ACKNOWLEDGEMENT

This work is supported by National High-Tech R&D (863) Program (no. 2015AA01A707) and National Science Foundation of China (no. 61272444, U1401253, U1405251, 61411146001).

REFERENCES

[1] Newsweek, "Hundreds of intimate celebrity pictures leaked online following alleged icloud breach," *http://www.newsweek.com/hundreds-intimate-celebrity-pictures-leaked-online-following-suspected-icloud-267851*, vol. 20, no. 3, pp. 273–297, 2014.

[2] I. D. Corporation, "Smartphone os market share," *http://www.idc.com/getdoc.jsp?containerId=prUS25450615*, Q4 2014.

[3] Z. W. M. Grace, Y. Zhou and X. Jiang, "Systematic detection of capability leaks in stock android smartphones," *Proceedings of the 19th Annual Symposium on Network and Distributed System Security, 2012.*, 2012.

[4] A. Shamir, "How to share a secret," *Communications of the ACM 22.11 (1979): 612-613.*, vol. 20, no. 3, pp. 273–297, 1979.

[5] A.Shamir and M.Naor, "Visual cryptography," *Proceedingsof the Advances in Cryptology-Eurocrypt '94*, 1995.

[6] T.-H. Chen and C.-S. Wu, "Efficient multi-secret image sharing based onboolean operations," *Signal Processing 91.1(2011) 90–97*, 2011.

[7] N. Askari, C. Moloney, and H. Heys, "A novel visual secret sharing scheme without image size expansion," in *Electrical & Computer Engineering (CCECE), 2012 25th IEEE Canadian Conference on*, 2012, pp. 1–4.

[8] C.-C. C. Cheng Guo and C. Qin, "A hierarchical threshold secret image sharing," *Pattern Recognition Letters*, 2011.

[9] Q. L. Xiaotian Wua, Duanhao Oua and W. Sun., "A user-friendly secret image sharing scheme with reversible steganography based on cellular automata," *The Journal of Systems and Software*, 2012.

[10] W. Lou, W. Liu, and Y. Fang, "Spread: enhancing data confidentiality in mobile ad hoc networks," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 4, 2004, pp. 2404–2413.

[11] G. BLAKLEY, "Safeguarding cryptographic keys," *Proc. AFIPS 1979 National Computer Conference. Vol. 48.*, vol. 30, pp. 50–64, 1979.

[12] M. L. M. H. A. Narayanan, N. Thiagarajan and D. Boneh, "Location privacy via private proximity testing," in *In proceedings of NDSS 2011.*, 2011, pp. 381–386.

[13] Z. Zhu and G. Cao, "Applaus: A privacy-preserving location proof updating system for location-based services," in *INFOCOM, 2011 Proceedings IEEE*, 2011, pp. 1889–1897.

[14] S. L. P. E. Di Qiu, Dan Boneh, "Robust location tag generation from noisy location data for security applications," *Proceedings of the 2009 International Technical Meeting of The Institute of Navigation*, 2009.

[15] W. L. Y. T. H. Yao Zheng, Ming Li, "Sharp: Private proximity test and secure handshake with cheat-proof location tags," *ESORICS 2012*, vol. 6, no. 10, pp. 818–823, 2012.

[16] P. E. SHERMAN LO, DAVID DE LORENZO, "Signal authentication - a secure civil gnss for today," *Inside GNSS*, vol. 327, no. 5968, pp. 1018–1021, 2009.