International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.9 (2015), pp.155-166
http://dx.doi.org/10.14257/ijmue.2015.10.9.17

K-Anonymity Algorithm Using Encryption for Location Privacy Protection

Jinying Jia, Fengli Zhang

School of Computer Science & Engineering, University of Electronic Science and Technology of China, Chengdu, China
jiajinying@126.com

Abstract

In this paper, we solved a location privacy protection in location-based services (LBS) where the mobile user had to report her exact location information to an LBS provider for the purpose of obtaining her wished services. Location invisible had been well proposed and researched to defend user privacy. However, as the nature of the insecure wireless net environment the user's location information may still be uncovered. We presented a k-anonymity algorithm using encryption for location privacy protection that can improve security of LBS system by encrypting the information transmitted by wireless. The experimental results evidence that the processing time is acceptable in current low-level devices.

***Keywords:** location based service; location privacy; spatial cloaking; k-anonymity*

1. Introduction

A booming sale of smart mobile equipments is witnessed by the consumer electronics markets in recent years. These equipments, such as smart phones and Personal Digital Assistants (PDA), are has powerful CPU, large ROM and RAM, positioning technology (*e.g.*, GPS and AGPS). New applications for the mobile users are opened up by the omnipotence of these equipments [1-6]. In particular, the mobile users can enjoy mobile online inquiry in LBS with the combination of GPS and wireless internet. The LBS provides dynamic content according to where the user is located. Typical applications of the mobile online inquiry in LBS include the nearest point of interest (POI) query, location-aware advertisement, and road navigation. In order to inquire online, the mobile users must explicitly expose their precise locations to the server. For example, if a mobile user inquires her nearest hospital, she must provide her precise position in terms of GPS coordinates to the LBS server. In this sense, the privacy of the user's location is compromised in exchange for services. Cache the entire dataset of POI on the mobile equipment is an intuitive way to solve this problem. It can resolve location-based queries locally, but the resources of the mobile device are limited. This method cannot deal with data updates; neither can scale to large POI datasets. Therefore, location anonymity has been well researched. The location cloaking allows the mobile user to inquire online without exposing the precise location [2-4,7-10]. It uses an anonymous spatial region (ASR) instead of the user's precise location. The ASR must include at least k users and the size of the cloaked region must overstep a threshold. Nevertheless, the user's location information may still be exposed as the nature of the unreliable wireless network environment for all existing algorithms.

Figure 1 depicts an example for eavesdropping. If the user uses the system of location privacy protection with the assailant's AP, the assailant can get the user's location information easily.

ISSN: 1975-0080 IJMUE

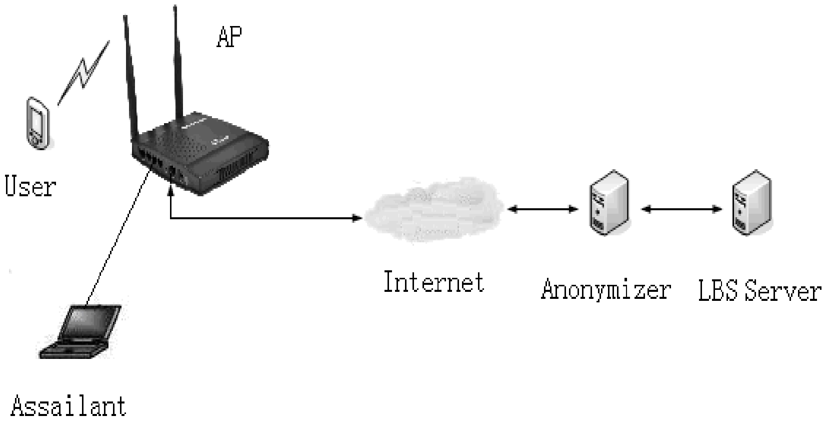


Figure 1. Example for Eavesdropping

In this paper, we proposed a k-anonymity algorithm using encryption for Location Privacy Protection. It can improve security of LBS system by using our combination of encryption scheme to encrypt the information transmitted by a wireless network.

To summarize, our contributions in this paper are as follows:

- Proposed a combination of encryption scheme for k-anonymity of location privacy protection. We used EIGamal [13] public-key cryptosystems to encrypte the secret keys for the users and the centralized anonymizer. After that, we used the Advanced Encryption Standard (AES) symmetric cryptosystems with the secret keys to encrypt the information transmitted by a wireless network.
- Proposed a segmented encryption scheme for the AES keys using EIGamal. The AES Key was divided every 8 bits, each segment was encrypted using EIGamal.

The rest of the paper keeps on as follows. Section 2 reviews existent work on location anonymity. Section 3 presents the system architecture. Section 4 presents some algorithms for the system. Section 5 presents the experimental results. The conclusion and future work are shown in Section 6.

2. Related Work

Location incognito has attracted intensive study as a solution to preserve user privacy in mobile computing, particularly for LBS. The aim is to permit a mobile user to apply for services without disclosing her exact location. In all kinds of anonymous techniques, location invisible is the predominant. It transmits to the server a cloaked region (such as a rectangle or a circle) that includes the user’s exact location and is big enough to satisfy some privacy metric. The two most diffusely applied metrics are k-anonymity — this region must include at least k users, the real requesting user is indistinct from at least k-1 other mobile users who are in the same cloaked region, and granularity —this region must surpass a threshold. Interval Cloak [8] is one of the first cloak techniques. The anonymizer uses a quad-tree to index the mobile users. To produce an ASR for the querying user, Interval Cloak searches the quad-tree from down to the topmost node that includes at least k users (included request user). The extent of this node is come back as the ASR. Casper Cloak [9] resembles to Interval Cloak, with two primary differences. First, Casper Cloak determines and accesses the quad-tree’s the leaf immediately through utilize of a hash table. Second, instead of directly back tailing to the parent quadrant, it verifies the two adjoining quadrants to find if their combination with the user quadrant includes k users. Gedik and Liu proposed a customized k-anonymity model named

Clique-Cloak, it builds a clique graph to joint clients that can share the same cloaked region [10]. They addressed the problem when a client continuously requests location indiscernible, and proposed an optimal cloaking algorithm to fight tracing analysis attacks.

Yingjie Wu *et al.* proposed a Guess-Answer cloaking algorithm to generate ASR with the interaction between user and server [11]. The server guesses an ASR, and gives it to the user. The user tells her relative direction of the guess ASR to the server. The server guesses a new ASR with the answer, and gives it to the user. The user answers it. This work continues until the user is in the guessed ASR. Their algorithm can work without the third centralized anonymizer. Haibo Hu and Jianliang Xu proposed a non-exposure cloaking algorithm to generate the ASR. Their algorithm is carried out using the proximity information among mobile users, not directly using their accurate coordinates [12].

3. System Architecture

Figure 2 depicts the system architecture that consists of three entities, mobile users, anonymizer and LBS server. We will first discuss our privacy threat model, and then describe privacy settings in user privacy profiles, and each entity in our system.

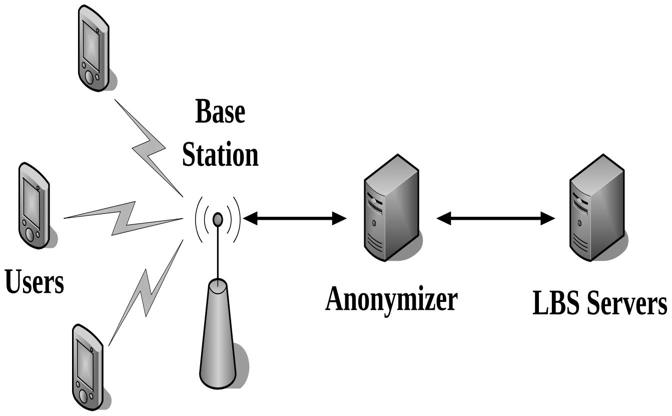


Figure 2. System Architecture

Privacy threat model we assume that the centralized anonymizer is trusted. However, LBS providers (LBS Servers) are not trusted.

User privacy profiles each user specifies her privacy requirements in a privacy profile in a form of (A_{min}, K). The required minimum area of her ASR is indicated by A_{min}. The required anonymity level is indicated by K. The inquiry user can alter her privacy profile at the initiation of any query is very important. It can guarantee that the query user specified privacy settings achieve her wished privacy protection in different situations.

Mobile users each mobile user has a wireless network for communicating with the anonymizer, *e.g.*, GPRS, WCDMA, and CDMA2000, and a GPS or AGPS equipment, to determine her location which can be represented as a coordinate (x, y).

Anonymizer the anonymizer has a table to record every user’s AES key, and a quad-tree to record every user’s coordinate.

LBS servers LBS servers have a privacy-aware query processor, which has the ability to deal with location-based queries with ASR for one query. It can only calculate a candidate set (CS) of points of interest (POI) for all the area in the ASR which includes the exact answer to the user, because the query processor does not know the exact user location.

4. Algorithms

In this Section, we give the algorithms of the ElGamal cryptosystem, and the algorithms of our system.

Algorithm 1: Key generation for EIGamal encryption

```
function produceEIGamalKey(message m )
    Generates a large random prime p using m;
    Generates the generator g of $Z_p^*$ using m;
    Chooses a random integer k, $1 \leq k \leq p-2$;
    Calculates $g^k \bmod p$;
    return public_key (p, g, $g^k$)& private_key (k);
```

Algorithm Algorithm 1 depicts the pseudo code of the Key generation for EIGamal public-key encryption. At first of all, it generates a large random prime p and the generator g of Z_p^* using the message m. And then, it chooses a random integer k ($1\leq k \leq p-2$), and calculates "g^k mod p". In the end, it returns the **(p, g, g^k)** as the public key and **(k)** as the private key.

Algorithm 2: EIGamal public-key encryption

```
function encrypt ($\mathbf{m, p, g, g^k}$)
    Represents the message m as an integer m in the range {0, 1, ... ,p-1};
    Chooses a random integer r, $2 \leq \mathbf{r} \leq p-2$;
    Calculates $\mathbf{c_1} = g^{\mathbf{r}} \bmod p$;
    Calculates $\mathbf{c_2} = \mathbf{m} \cdot (g^{\mathbf{k}})^{\mathbf{r}} \bmod p$;
    return cipher text $\mathbf{c} = (\mathbf{c_1}, \mathbf{c_2})$;
```

Algorithm Algorithm 2 depicts the pseudo code of the EIGamal public-key encryption. At first of all, it represents the message m as an integer m in the range {0, 1, ... ,p-1}. And then, it chooses a random integer r ($1\leq r \leq p-2$), and calculates "$\mathbf{c_1} = g^r \bmod p$" and "$\mathbf{c_2} = \mathbf{m}\cdot (g^{\mathbf{k}})^{\mathbf{r}} \bmod p$". In the end, it returns the ($\mathbf{c_1}$, $\mathbf{c_2}$) as the cipher text.

Algorithm 3: EIGamal public-key decryption

```
function decrypt ($\mathbf{k, c_1, c_2}$)
    Calculates $\mathbf{c_1}^{-\mathbf{k}} = \mathbf{c_1}^{p-1-\mathbf{k}} \bmod p$;
    Calculates $\mathbf{m} = \mathbf{c_1}^{-\mathbf{k}} \cdot \mathbf{c_2} \bmod p$;
    return message m;
```

Algorithm Algorithm 3 depicts the pseudo code of the EIGamal public-key decryption. At first of all, it uses the private key **k** to calculate "$\mathbf{c_1}^{-k} = \mathbf{c_1}^{p-1-\mathbf{k}} \bmod p$". And then, it calculates "$\mathbf{m} = \mathbf{c_1}^{-k} \cdot \mathbf{c_2} \bmod p$". In the end, it returns the **m** as the original text.

Algorithm 4: Initialization of the mobile user

```
function userInitialize( )
    Receives the anonymizer public keys ($\mathbf{p, g, g^k}$);
    Creates and preserves her AES key $\mathbf{K_{aes}}$;
    Divides the $\mathbf{K_{aes}}$ every 8 bits, encrypts each segment using ($\mathbf{p, g, g^k}$); //call Algorithm 2.
    Sends her cipher text of all the segments to the anonymizer;
    Sends her new location AES(x,y) to the anonymizer at intervals of △t;
```

Algorithm Algorithm 4 depicts the pseudo code of the initialization of the mobile user. At first of all, she receives the anonymizer's EIGamal public key, creates and preserves her AES key. And then, she divides the AES key every 8 bits and encrypts each segment using the anonymizer's EIGamal public key, sends the cipher text of all the segments to the anonymizer. to the anonymizer. In the end, she sends her new location (x, y) encrypted using the AES key to the anonymizer at intervals of△t.

Algorithm 5: Initialization of the anonymizer

```
function anonymizerInitialize( )
  produceEIGamalKey( );  //call Algorithm 1.
  Sends her public key ($\mathbf{p, g, g^k}$) to the each user;
  Receives the cipher text of the $K_{aes}$ from each user;
  decrypt each segment of the $K_{aes}$ for each user using $\mathbf{k}$;  //call Algorithm 3.
  Preserves original text $\mathbf{m}$ of $\mathbf{K_{aes}}$ for each user;
  Repeatedly receives mobile users' new location AES(x,y), uses $\mathbf{K_{aes}}$ to decrypt the mobile users' new locations (x,y), updates the quad-tree using the new locations (x,y);
```

Algorithm Algorithm 5 depicts the pseudo code of the initialization of the anonymizer. At first of all, it calls Algorithm 1 to produce the EIGamal public key and EIGamal private key, and sends the EIGamal public key to each user. And then, for each user: it receives the cipher texts of the AES key, and calls Algorithm 3 to get the original texts of the AES key using its EIGamal private key for each segment, saves the AES key. In the end, it repeatedly receives the mobile user's new location encrypted using the user's AES key, uses the user's AES key to decrypt it to get the user's new location, update the quad-tree using the new location.

Algorithm 6: Process inquiry at the mobile user

```
function userProcessInquiry ( )
  Sends the cipher texts of her new location and her content of inquiry encrypted by her $\mathbf{K_{aes}}$ to the anonymizer;
  Receives the result comes back from the anonymizer;
  decrypts the result received by her $\mathbf{K_{aes}}$ to get the last result;
```

Algorithm Algorithm 6 depicts the pseudo code of the process inquiry at the mobile user. At first of all, she sends the cipher texts of her new location and her content of inquiry encrypted by her AES key to the anonymizer. And then, she receives the result comes back from the anonymizer. In the end, she decrypts the result received by her AES key to get the last result.

Algorithm 7: Process inquiry at the anonymizer

```
function anonymizerProcessInquiry ( )
  Receives the user's cipher texts of new location and her inquiry content;
  Decrypts the user's cipher texts using the user's $\mathbf{K_{aes}}$;
  Generates the ASR for the user using the quad-tree;
  Sends the ASR and the inquiry content to the LBS server;
  Receives the result comes back from the LBS server and removes the useless POIs by the user's location;
  Encrypts the result using the user's $\mathbf{K_{aes}}$ and sends it to the inquiry user;
```

Algorithm Algorithm 7 depicts the pseudo code of the process inquiry at the anonymizer. At first of all, it decrypts the cipher texts which are received from the user using the user's AES key. And then, it generates the ASR for the user using the quad-tree, sends the ASR and the inquiry content to the LBS server. In the end, it receives the result comes back from the LBS server, removes the useless POIs by the user's location, and send the last result encrypted using the inquiry user's AES key to the inquiry user.

5. Experiment

5.1. Feasibility Experiment

Our purpose is to improve the security for the k-anonymity by the AES and ElGamal cryptosystem, so we just concern about whether our combined scheme from the AES and ElGamal cryptosystem is suitable for k-anonymity mobile online inquiry in LBS. In the experiment, we use a mobile phone with "CPU 600Hz, RAM 256M and OS Android 2.3" to measure the time for running the algorithms of the ElGamal cryptosystem.

Figure 3 depicts the time for running the algorithms on the mobile phone, the message used in the experiment for ElGamal encryption is "12345678" which is used as the user's AES key (The program will add 0 after the key, let its length be 128 bits. AES encryption and AES decryption is "(N 48°12.511′, E 016°22.379′)". The key for ElGamal is (p=8847727541916506459, g=2944959510029594611, g^k=587565072995521 4451). We do the same experiment for 100 times. The maximum time of the ElGamal is 22,186,668 nanoseconds; the minimum time of the ElGamal is 851,667 nanoseconds; the mean time of the ElGamal is 1,470,867 nanoseconds. The maximum time of the AES is 11,540,000 nanoseconds; the minimum time of the AES is 801,667 nanoseconds; the mean time of the AES is 1,307,816 nanoseconds.

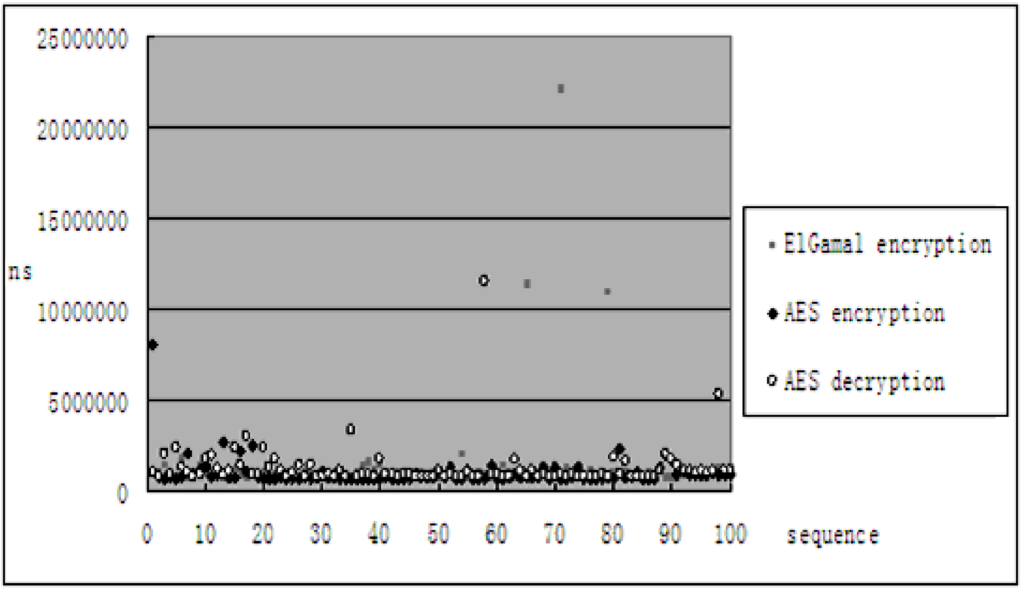


Figure 3. Time for Running the Algorithms on the Mobile Equipment

We can see the time does not exceed 23 milliseconds, and most are between 1 milliseconds to 4 milliseconds from Figure 2. The mobile phone used in the experiment belongs to the low configuration, today there are a lot of mobile phones with "CPU 1024HZ RAM 2048M", so the time for running our combined scheme from the AES and ElGamal cryptosystem on the mobile phones is suitable for mobile online inquiry in LBS.

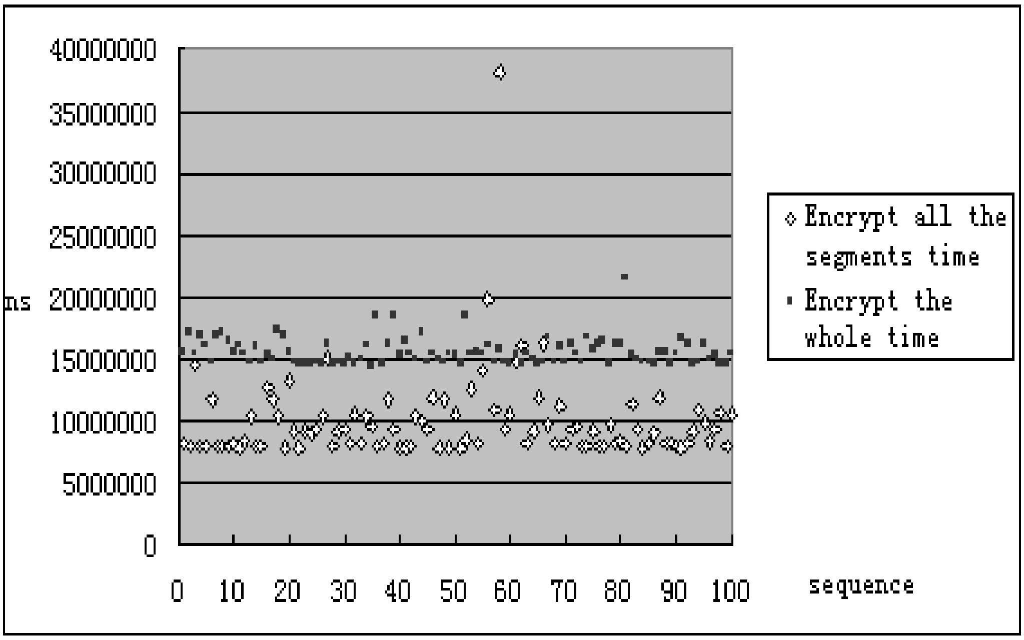


Figure 4. Time for Encrypting the AES Key by different Methods

Figure 4 depicts the time for encrypting the AES key by different methods on the mobile phone, the message used in the experiment for ElGamal encryption is 8 “12345678” which are linked one by one. The total time of our method for encrypting all the segments is just half of the time for encrypting the whole AES key without dividing. When the AES key is 128 bits, the time of our method is same as the time for encrypting the whole 64 bits AES key without dividing. However, the time for encrypting the whole 128 bits AES key without dividing is longer than 10 minutes. So our method for encrypting the AES key using ElGamal is suitable for k-anonymity location privacy protection in LBS.

Figure 5 depicts the time for running the DES, ElGamal, RSA and AES encryption on the mobile phone, the messages used in the experiment for ElGamal encryption and AES encryption are both “(N 48°12.511′, E 016°22.379′)”. The key used for DES and AES is “12345678”. The key used for ElGamal is (p=491131371833358229191846096174448531605064987790728038308627,g=316375794459409284107890071599924407054488454054823332501207,g^k=50671560519853688457833279153078104223766647407690845692020). The key used for RSA is MIGfMA0GC SqGSIb3DQEB AQUAA4GNADCBiQKBgQCvws3Zx2SJaTFoIL87/EN6f7xQobulcTEhMwvNNo8 Ebxl+mze59ac8fQiMT1dk1HptwBpwoRYbA3d9/VyJJUjKydeYJNo7l34JoGzKZYN cRh7BC6YWgP6XCOuHP6TfOeEEfe73s2jMVpG/1qKlzzh2UnNuchCc0nAWjsQ3T yeEmwIDAQAB. We can see the encryption time of ElGamal is shorter than RSA, and the encryption time of AES is shortest.

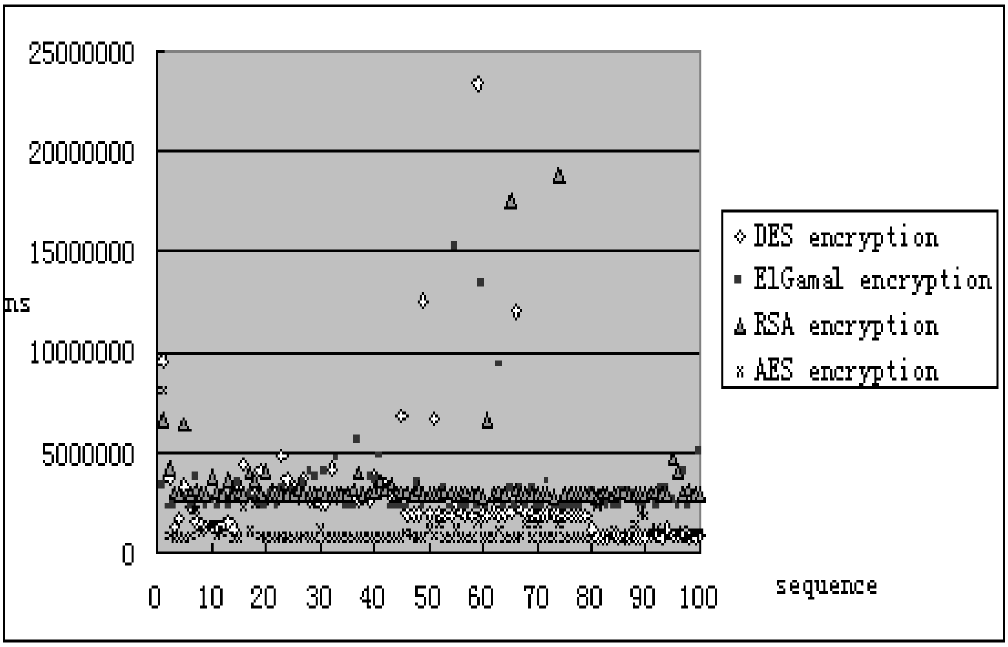


Figure 5. Time for Running the DES, ElGamal, RSA and AES Encryption

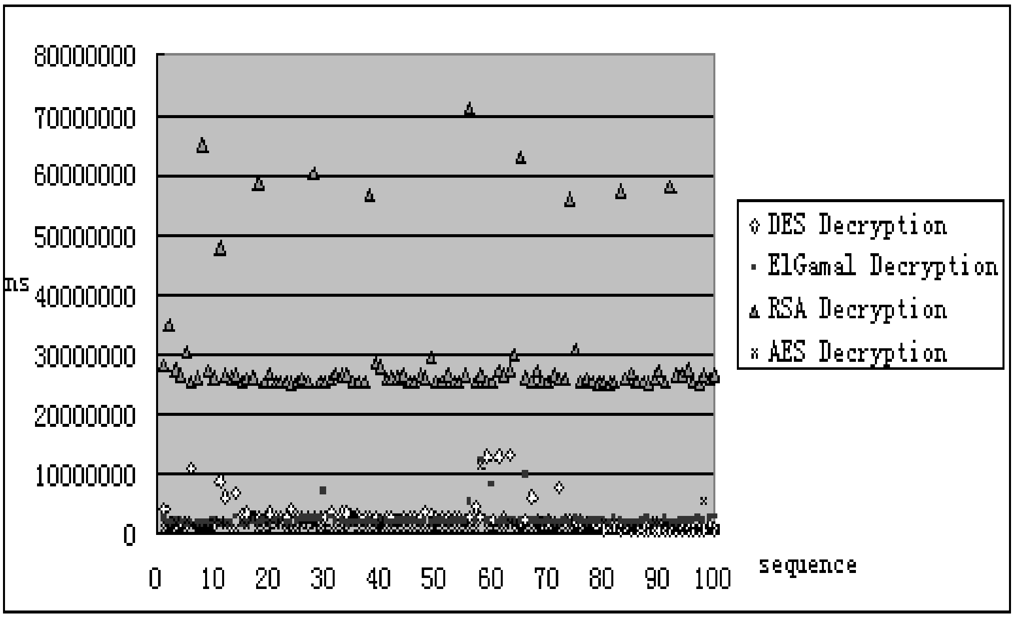


Figure 6. Time for Running the DES, ElGamal, RSA and AES Decryption

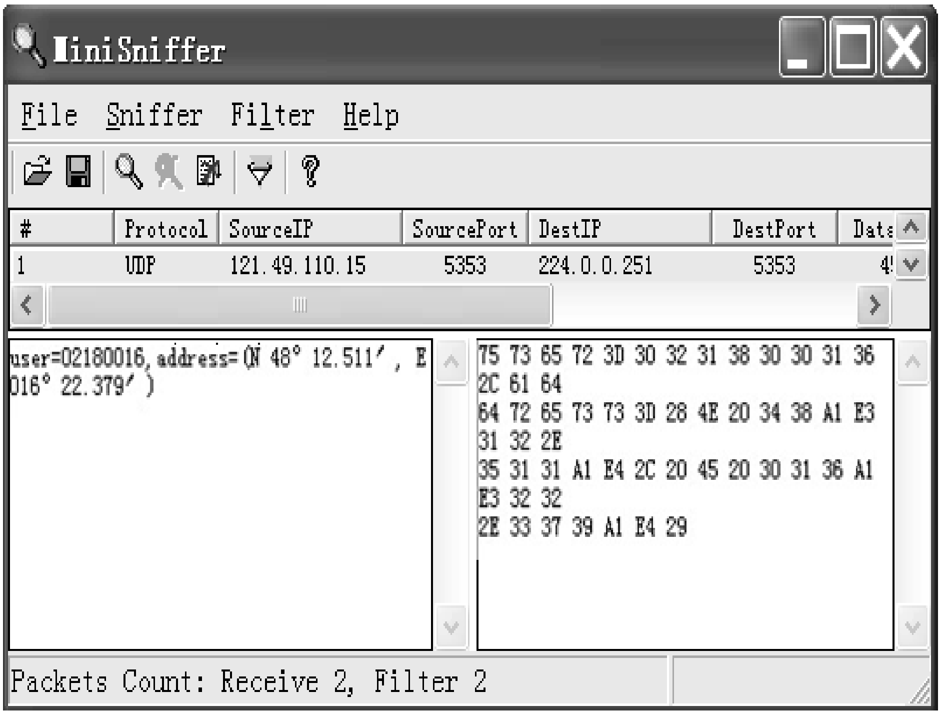
Figure 6 depicts the time for running the DES, ElGamal, RSA and AES decryption on the mobile phone, the messages used in the experiment for ElGamal encryption and AES encryption are both the cipher text of "(N 48°12.511′, E 016°22.379′)". The key used for DES and AES is "12345678". The key used for ElGamal is 275876583916262184 779941024359967270120625132378215345558390. The key used for RSA is MIICdwIBADANBgkqhkiG9w0BAQEFAASCAmEwggJdAgEAAoGBAK/CzdnHZi
lpMWggvzv8Q3p/vFChu6VxMSEzC802jwRvGX6bN7n1pzx9CIxPV2TUem3AGnC
hFhsDd339XIklSMrJ15gk2juXfgmgbMplg1xGHsELphaA/pcI64c/pN854QR97veza
MxWkb/WoqXPOHZSc25yEJzScBaOxDdPJ4SbAgMBAAECgYEAiC09wsMrUQ/
QuOXOZRKR0aKQbESzF3742skrFqdz7bKKpT5r0cfT+BjD+7opczTxWoFhuuwEL

Y/lEBeyj4v0teeu4XiCRDFmyuQPXI217JagQ76gZfkSV6C/4EWJCx5b7KW2/v1mV
jjRSt5xqKUjS9QTe/n7kzGk1y8dkairnGECQQDphrvvQp7imDEZdW5rwpcJ35r0JR
7O/+GEFzu2rqZvG4WX8H5GWicEZW0oCrUUSEEKvxpmUaue3pLi8K7KHN1LA
kEAwKzvZA/+QRN2EcwF7zQWL7bAA5ap733PhfN2YxEw/uMGTcxlppDRcJERH
dwxnL6ArRco2L3h7yh2WKkS2Kfz8QJBAIDAdTt8Cqe561AzC5dEKQwbiY5ULQ
FJ2Od/+79D2aVmkihsQuK878gft6gdBytQjvPC22ZQXwPviSKwOl5avbMCQDFn5t
bMtxCu0dbMbzUgMRQ5E1GYP6kWpLEfddr/XIXvZv1qDSLB/cKUoEP844fi3ZIE
30d8oUO3J2miqR9/KbECQH21ksHbNPBem0J8jifk52BOj76uwY73wBokTJIE30Ki
dDNWKJGTf9W4sb05YvQn5S2yUf4OYvP4QuIbyH74TSQ=.

We can see the encryption time of ElGamal is shorter than RSA, and the encryption time of AES is shortest from Figure 5 and Figure 6. So we use the AES and the ElGamal.

5.2 Security Experiment

In order to test and verify whether our algorithm is safer than the classical algorithm [7], we use MiniSniffer to eavesdrop the wireless net.



#	Protocol	SourceIP	SourcePort	DestIP	DestPort	Data
1	UDP	121.49.110.15	5353	224.0.0.251	5353	4!

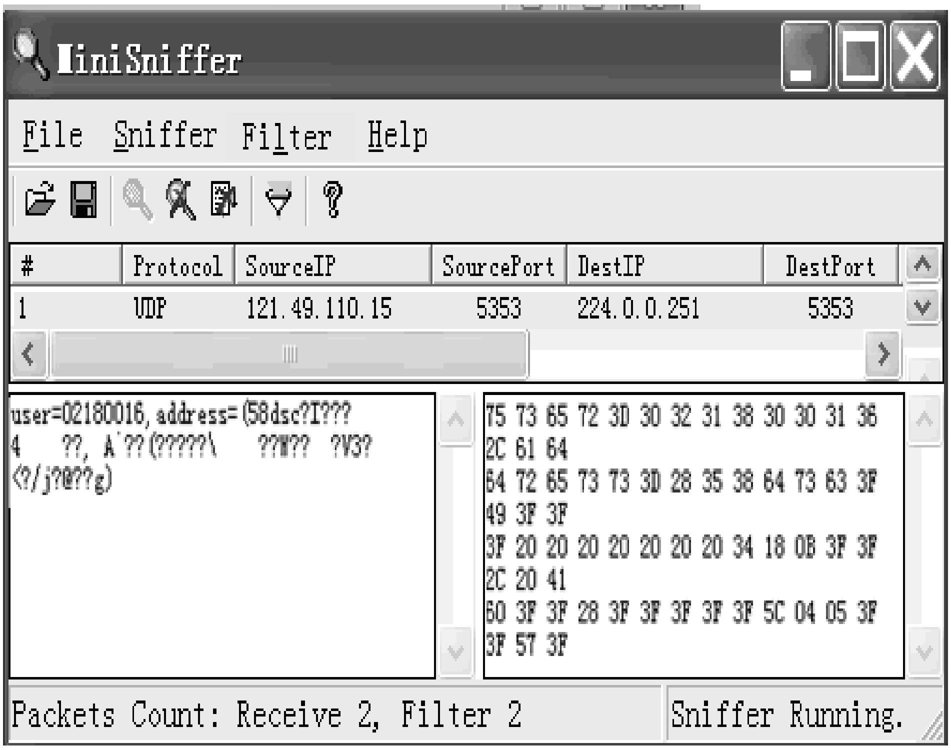
user=02180016,address=(N 48° 12.511′ , E 016° 22.379′)

```
75 73 65 72 3D 30 32 31 38 30 30 31 36
2C 61 64
64 72 65 73 73 3D 28 4E 20 34 38 A1 E3
31 32 2E
35 31 31 A1 E4 2C 20 45 20 30 31 36 A1
E3 32 32
2E 33 37 39 A1 E4 29
```

Packets Count: Receive 2, Filter 2

Figure 7. Eavesdropping Experiment for Classical Algorithm

Figure 7 depicts the result of the eavesdropping experiment for classical algorithm. We can see the 02180016 user’s location is (N 48°12.511′, E 016°22.379′).



#	Protocol	SourceIP	SourcePort	DestIP	DestPort
1	UDP	121.49.110.15	5353	224.0.0.251	5353

```
user=02180016,address=(58dsc?I???
4    ??, A`??(?????\    ??W??  ?V3?
<?/j?@??g)
```

```
75 73 65 72 3D 30 32 31 38 30 30 31 36
2C 61 64
64 72 65 73 73 3D 28 35 38 64 73 63 3F
49 3F 3F
3F 20 20 20 20 20 20 20 34 18 0B 3F 3F
2C 20 41
60 3F 3F 28 3F 3F 3F 3F 3F 5C 04 05 3F
3F 57 3F
```

Packets Count: Receive 2, Filter 2

Sniffer Running.

Figure 8. Eavesdropping Experiment for our Algorithm

Figure 8 depicts the result of the eavesdropping experiment for our algorithm. We cannot get the 02180016 user’s location, because what we get is the cipher text of the user’s location.

We can see that our algorithm is safer than the classical algorithm from Figure 7 and Figure 8.

6. Conclusions

In general, we presented a k-anonymity algorithm using encryption for location privacy protection in LBS, which can improve security of LBS system by using our combined scheme from the AES and ElGamal cryptosystem to protect the information transmitted by wireless network, and proved the time for running our combined scheme from the AES and ElGamal cryptosystem on the mobile phones is suitable for k-anonymity location privacy protection in LBS by experiment.

Acknowledgements

This work is supported in part by Important National Science and Technology Special Project of China (No. 20112X03002-002-03). This work is also supported partly by the National Nature Science Foundation of China under Grant (No. 60903157 and No. 61133016), and the National High Technology Joint Research Program of China (863 Program, Grant No. 2011AA010706).

References

[1] J. Y. Jia and F. L. Zhang, “Overview of location privacy protection technology”, Application Research of Computers, vol. 30, no. 3, **(2013)**.

[2] J. Y. Jia and F. L. Zhang, “Twice Anonymity Algorithm for LBS in Mobile P2P Environment”, Journal of Computational Information Systems, vol. 9, no. 9, **(2013)**.

[3] J. Y. Jia and F. L. Zhang, "Non-exposure Accurate Location K-Anonymity Algorithm in LBS", Scientific World Journal, doi: 10.1155/2014/619357, **(2014)**.

[4] J. Y. Jia and F. L. Zhang, "An Incremented KNN Inquiry Algorithm Based on the Grid of Latitude-longitude for Location Privacy Protection", Application Research of Computers, vol. 31, no. 12, **(2014)**.

[5] J. F. Feng and W. Wei, "Multiple spatial model fusion in heterogeneous sensor networks", International Journal of Multimedia and Ubiquitous Engineering. vol. 9, no. 2, **(2014)**.

[6] Y. Seo and J. Ahn, "Novel method for enhancing contents recommendation accuracy using LBS-based users viewing path similarity", International Journal of Multimedia and Ubiquitous Engineering, vol. 8, no. 4, **(2013)**.

[7] C. Y. Chow, M. F. Mokbel and J. B. Xuan, "Query-aware location anonymization for road networks", Geoinformatica, vol. 15, no. 3, **(2011)**.

[8] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking" Proc. of MobiSys; San Francisco, **(2003)**.

[9] M. F. Mokbel, C. Y. Chow and W. G. Aref, "The new Casper: Query processing for location services without compromising privacy", ACM Transactions on Database Systems, vol. 34, no. 4, **(2006)**.

[10] P. Kalnis, G. Ghinita and K. Mouratidis, "Preventing location-based identity inference in anonymous spatial queries", IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 12, **(2007)**.

[11] Y. J. Wu, S. S. Zhong and X. D. Wang, "Guess-answer: Protecting location privacy in location based services", Advances in Intelligent and Soft Computing, Shanghai, December 1-3, **(2011)**.

[12] H. B. Hu and J. L. Xu, "Non-Exposure Location Anonymity", In: IEEE International Conference on Data Engineering, Shanghai, China, March 29-30, **(2009)**.

[13] S. Prashant, S. Sonal and D. R. Shankar, "Modified ElGamal Cryptosystem Algorithm", The International Conference on Computer and Communication Technology, Allahabad, India, September 15-17, **(2011)**.

Authors

Jinying Jia, male, born on 1982, Han nationality, PhD students in the School of Computer Science & Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China (e-mail: jiajinying@126.com). His interests include: LBS, GIS and Information Security.

Fengli Zhang, female, born on 1963, Han nationality, Doctor, Professor, a tutor for doctors, works at University of Electronic Science and Technology School of Computer Science, Her interests include: Computer Applications and Information Security.