Journal of Science and Technology Volume 48, Issue 4, 2010 pp. 141-152

HANGOUTPLUS: A PRIVACY PRESERVING SOCIAL NETWORKING SERVICE PROVIDING REAL-TIME AND MORE SECURE PROTOCOL BASED ON HANGOUT SYSTEM

TRAN HONG NGOC

ABSTRACT

HangOut is a privacy preserving location based social networking service proposed by Annavaram, Jacobson and Shen in [1]. This system helps protect users' private information against malicious users and even administrators, based on two key ideas as: *anonymous update* and *density request*. It uses location and time distortions, AES and RSA in encrypting the shared key and protecting the data on the communications. Hence the protocol in this system has four issues: the first issue is the limit in realtime attribute, since they used RSA for clients (mobile devices) to encrypt data transmitted to server. RSA is an assymmetric cryptosystem based on big prime numbers on which calculative operations perform slowly and need more hardware resources. The second one is that HangOut protocol cannot be against active attackers to find out links between client and its location records in database because of the way it updates and replies density requests. The third one is its unsecure key management and protocol, the shared key is kept on clients and they share key between peer-to-peer, there is no authenticating clients, so clients can be made fake. The four one is that the ability of the database server decrypt and can read the message which can be intervened by active attackers. In this paper, we proposed methods to improve HangOut System, and to help it improve its realtime and security in preserving the privacy, the proposed method is named HangOutPlus.

Keywords. location privacy, *k*-anonymity, location privacy protocol, mobile network, social network, mobile social network, social network application.

1. INTRODUCTION

Social networks nowadays, such as facebook, twitter, myspace, hifi, etc., have grown up with the high speed by more and more tremendous number of users [2]. And in the current modern active society people always travel, which is the reason for the strong improvement and development progress of mobile devices, mobile networks and social network applications running on mobile devices. Moreover, the most interesting is finding the fact that more people are using the mobile web to socialize (91%) compared to the 79% of desktop users who do the same. It appears that the mobile phone is actually a better platform for social networking than the PC [3], and during May 2010, social networks accounted for 11.88% of UK Internet visits and search engines accounted for 11.33%, representing the first ever month that social networks

have been more popular than search engines in the UK [4]. By that ratio, we can see the number of mobile social network application users is increasing rapidly.

Actually, since people often travel to very far places, users can use mobile devices for convenience and through them to access their favorite social networks and then to communicate with their friends or to observe their friend's present situation. Besides, through social networks users can share information to the others which is friends, relations, or even strangers, etc. However, social network users on mobile networks accidently reveal their private or location information to the others and they may be used for many illegal targets. So there have been a lot of solutions to support social network users limit in revealing their sensitive information (or data) such as policies, but they are ineffective. And authors in [1] proposed HangOut which is a social networking application that allows its users to interact with each other without ever disclosing the precise location information of any single user. HangOut preserves participants' privacy in two fundamental ways, namely *anonymous updates* and *density requests*. It allows users to control their privacy while sharing their location with other participants. It also protects users against having their location tracked by anyone, including the service provider itself. HangOut ensures that there is no record of where users where that can be recreated from the system. However, HangOut still has some shortcomings: the first shortcoming is the limit in realtime attribute, since they used RSA for clients (mobile devices) to encrypt data transmitted to server. RSA is an assymmetric cryptosystem based on big prime numbers on which operations perform slowly and need more hardware resources. The second one is that HangOut protocol cannot be against active attackers to finf out links between client and its location records in database because of the way it updates and replies density requests. The third one is its unsecure key management and protocol, the shared key is kept on clients and they share key between peer-to-peer, there is no authentication clients, so clients can be made fake. These shortcomings are fixed in this article.

In this article, we strongly emphasis that we just present the proposed improved HangOut protocol for real-time and security attributes. We also introduce related works in section 2. In section 3, we describe HangOut system. We explain disadvantages of HangOut system in section 4. Then, we present the proposed improved method to solve these out comings by using the more secure key management and exchange protocols and making the system privacy higher by using dummy messages. In section 5, we describe the experiment and evaluation on our proposed solution. Lastly, we conclude and give the future works in section 6.

2. RELATED WORKS

Research users is paying attention to the privacy in mobile network applications, especially social network applications on mobile network which become more and more popular and the rise of this kind of applications is very dramatic. Up to now, there have been many researching works on privacy of the traditional social networks.

For the analysis of SNS disclosure problems, Gross et al., in 2005, analyzed more than 4,000 profiles from an SNS and found considerable disclosure of personal data [5]. Lewis et al. analyzed 1710 profiles of SNS users and found that all of these users had changed their default preferences [6]. Their analysis may imply that SNS users have become more concerned about privacy. It also means that users recognize the problems with existing disclosure controls.

For detection of private information revelation, Lam et al. developed a tool that detects private information in friend annotations [7]. For studying the role of social networks and social structure in facilitating criminal behavior, Antoni Calvó-Armengol and Yves Zenou in [8] proposed a model in which delinquents compete with each other in criminal activities.

For the privacy quantification, there has been one study on measuring the amount of private information leaked from profile information on SNSs [9].

For quantifying the privacy, such as the method uses probability and entropy theory [10]. There is one study about privacy quantification based on leaked data content posted on SNSs [11].

Regarding to location privacy metrics, Hoh and Gruteser [12] quantify location privacy as the expected error in the distance between a person's true location and an attacker's uncertain estimates of that location. Duckham and Kulik [13] define the "level of privacy" as the number of different location coordinates sent by a user with a single location-based query. In introducing k-anonymity to location privacy, Gruteser and Grunwald [14] use k to represent the level of privacy. Hoh et al. [15] quantify location privacy as the duration over which an attacker could track a subject. Some studies on computational location privacy show the relationship between location privacy and quality of service (QoS) [16, 17]. Sweeney's on the concept of *k*-anonymity for data privacy [18], and other privacy metrics include *t*-closeness [19] and *l*-diversity [20].

Till when the social network applications have been used on mobile networks, people have a concern on compromising privacy leading to serious security concerns. There are some research works on it, such as: Capra et al. [21] proposed a middleware architecture for providing privacy in mobile environments. Tang et al. [22] proposed a distributed method for storing personal information in mobile devices where personal information is split between mobile device and a trusted central server. Hoh et al. [23, 24] proposed a social network based traffic. C.A. Ardagna et al. [25] gave a multi-path approach for k-anonymity in mobile hybrid networks. Bhuvan Bamba et al. [26] give the anonymous location queries supporting in Mobile Environments with PrivacyGrid. And C.A. Ardagna et al. [27] also give the privacy preservation method over untrusted mobile networks.

3. HANGOUT SYSTEM

In this section, we resume HangOut system [1] architecture, its functionalities and properties, then we describe its shortcomings which are improved by proposed methods in the Sec. 4.

3.1. HangOut Overview

HangOut is a mobile service letting participants share their location information while preserving their privacy. In this article, we use two concepts "mobile device" and "client" as the same.

There are three user groups allowed in this system as: public groups, private groups, personal groups. Each group has the group number, the limit time and the group size.

- *Public groups* is open to anyone on network who is interested in joining the group. They can share similar interests one another. The information of this kind of group is available on mobile network. People can find them easily by using their mobile devices or through websites to search.

- *Private groups* is similar to the public group, but users needs authorization either from the group administrators who create a private group.
- *Personal groups* allows a small group of people to share their precise location for a finite period of time, is useful for a family or a group of friends to track each other.

HangOut uses GPS location. Each GPS location is a triple (X, Y, Zoom). X and Y are 25-bit value and are the offsets from the top left of the quadrant representation of world. Zoom belongs to the domain [1, 25], but at present HangOut can just support zoom value to 15.

HangOut uses four-tier architecture to protect privacy and provide the functionality (see Fig. 1). HangOut functions based on two key ideas: anonymous updates and density requests. *Anonymous update* means when a client moves to a new location, it will send a message of its new location to the location server. This server will save the new location down to the same old record in the database. The density of a location is defined as the number of times any member of group has sent a location update to the location server. *Density request* means the request of from a mobile client wanting to know the most popular locations for a given interest group initiates a density request. The requests typically contain the location which is specified as the top left corner of a quadrant that is of interest. The server accesses the database to find the cumulative count of updates to that location and sends that information back to the location server.

The figure 1 describes the communication protocol supported the whole Hangout system. Now we explain from the right component to the left one of HangOut system, as follows:

Mobile Client. The mobile client receives GPS coordinates and converts it to the triple (X, Y, Zoom) value of a location. If users want to join the public group, they must have the group number obtained from a web or mobile device based on a discovery tool. If users want to join the private or personal group, they must have more the group access key which is created based on current time and MAC address of the device as the seed or a random number generator and used to authenticate themselves to the location server for requesting services.

Database Server. The database server holds density information related to all groups that are active in the HangOut system. One table in the database is each group. A record for an user. He/she will update his/her location frequently. Each record of public group will have a vector (X, Y, and Zoom) as the location. Zoom is the value of the granularity of each location and belongs to the range [1, 25]. The database does not have the identification information of the mobile device that is sent to update the database.

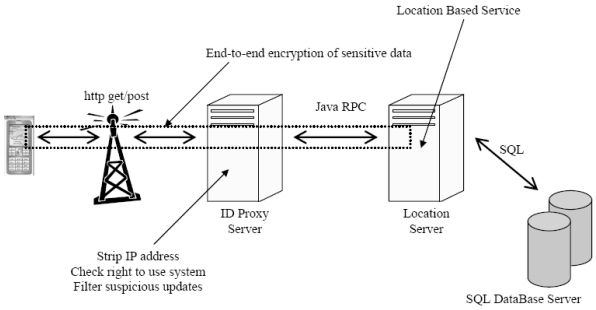

Figure 1. The four-tier architecture of HangOut system [1]

Location Server. The location server supplies services for mobile devices. HangOut uses RSA algorithm for encrypting messages from mobile devices to the location server. This server also publishes its public key for mobile devices to use for encrypting request messages before sending them to the location server. The messages encryption prevents adversaries from viewing the message content from mobile devices to the location server.

When the location server receives the encrypted message, it will decrypt the cipher text and get the request content. There are two requests: update request and density request. If the request is the update request, the location server will get the list of group from the message and scan in the database and overwrite the records it find out. If the request is the density request, the location server will calculate the relative density and return it to the mobile device.

ID Proxy Server. Mobile devices communicate with this server by sending and receiving HTTP messages**.** The message have 2 parts: the first part contains the mobile device identification information, such as phone number, MAC address, cell tower id of the original message. The second part changed based on the request type from the mobile device and what group type the request is intended for. There are two main kinds of request, they are anonymous updates and density request, beside creating new group request. When the server receives a message from a mobile device, it strips the first part from the message and transfer the second part to the location server. The location server does not access the mobile client identification information to request the service form the location server. And all request from the mobile device to the location server are encrypted by using the public key of the location server so that the proxy server and the third party cannot view the request content.

Besides, HangOut allows users send SMS messages to invite the other users to join the group, or for strange users to ask for permission to join the private or personal group. And HangOut also exchange the group private key for personal group invitations. But adversaties who listen to the SMS messages may correlate the SMS information and future message requests and potentially identify user location.

3.2. Privacy Persevering Methods in HangOut

HangOut proposed some privacy preservasing methods to prevent adversaries from viewing the private information and the location of mobile devices.

1. HangOut use RSA for encrypting all messages sent from mobile devices to the location server or vice versa, which prevents adversaries or the ID proxy server read the message content.
2. HangOut has 3 groups: private group, personal group, and public group. Users in each group will have different privacy level.

   With the public group, there is no need to give user's personal information when they join this kind of group. This group is open to anybody having the similiar interests. If users join this group and share their information which can be read by the strangers. So, user's private information is depending on their decision on sharing it or not. This kind of group has the lowest privacy level.

   With the private group and the personal group, all group members must have the group key to encrypt the location information or any personal message. So the messages are only read by the group members having the group key. The group creator will input the group

key or the group key is auto-generated. The group key is stored locally in the mobile device of the group creator. Using the group access key (using AES) prevents the adversaries from viewing messages sent among that group members. The kind of this group have the higher privacy level.

3. The ID Proxy Server strips the ID information of mobile device in the sent message so that the database server cannot recognize which message belongs to which mobile device, even after it decrypt the message. So if the database is compromised by adversaries, it makes them easy to link which client with which message, then they can know the group and the location to which mobile device belongs. Even they can find keys by using the statistics methods.
4. The database server does not have any identification information from the mobile device, since all messages sent to it are stripped the identification information of the mobile devices already, and the location needing updating will be overwritten on the old location entry in the database with the updated information. The database server just decrypt and view the content consisting of group number and member number. So there is no update track, even the encrypted format, available to adversaries. If adversaries can access the database and track all updates, they cannot read the message content.

4. THE PROPOSED HANGOUTPLUS

In this section, we analysis some issues of HangOut which affect to real-time and security of the system.

4.1. Issues in HangOut

In the section 3.2, we presented HangOut privacy abilities. And in this section, we presented some shortcomings affecting the privacy of HangOut system.

Issue 1. Performance and Real-time: Authors have suggested RSA as the security algorithm in HangOut. Mobile devices will use RSA algorithm to encrypt and decrypt all message transmitting between it and the location server. Actually, it is so slow since it uses big prime numbers and calculates on them. And in [28], the RSA key length that is asked for the security is 1024 bits (about 128 bytes) within the year 2010 and equal to or larger than 2018 bits (about 256 bytes) after the year 2010. So calculative operators on mobile devices have a lot of performance and resource limit.

Issue 2. Authentication: With the personal group and the private group, the group key is not changed during the group life cycle. And those keys are shared from peer to peer (means among mobile devices of the same personal group). So the authentication among users in a personal group or a private group are not ensured. Also with the public key of the location server, it can be make fake by an adversary or an intermediate. We need an effective key exchange and management protocol and the proper user authentication.

Issue 3. Anonymity: In HangOut, the way to store updated location can still reveal the location of mobile devices. If adversaries are active attackers, they can know which client just sent which request message by observing the relationship between each message sending events from clients and the update events in the database especially when there are a small number of clients functions in the system. Active attackers are the persons who can get the messages in the

system and modify them, besides observe links between clients and updated record in the database. So the way to do the update request in HangOut is really still not anonymous.

Issue 4. The location server cannot know which mobile device send or receive the message, except the ID proxy server. But if adversaries can compromise the ID proxy server, they can track which message links to which mobile device. The ability to decrypt the message of the database server is not secure. How to make the server be able to update correct entries in the database without decrypting the received messages?

4.2. Our approach for improving HangOut

We strongly suggest following solutions to fix the issues of HangOut demonstrated in the previous section.

Solution for issue 1. we use ECC algorithms instead of RSA algorithm. ECC is also the asymmetric cryptosystem as RSA, but it has the shorter key length than RSA's with the same security level. It can run faster than RSA and can save more time (see Table 2 [29]), energy and bandwidth. So it can ensure the real-time of the system. About the key strength between these two algorithm as the following table 1 and figure 2 [29].

Table 1. Key strength comparison between ECC and RSA.

RSA (bits)	ECC (bits)	Ratio	MISP years
1024	160	6:1	10^{12}
2048	224	9:1	10^{24}
3072	256	12:1	10^{28}
7680	384	20:1	10^{47}
15360	521	30:1	10^{66}

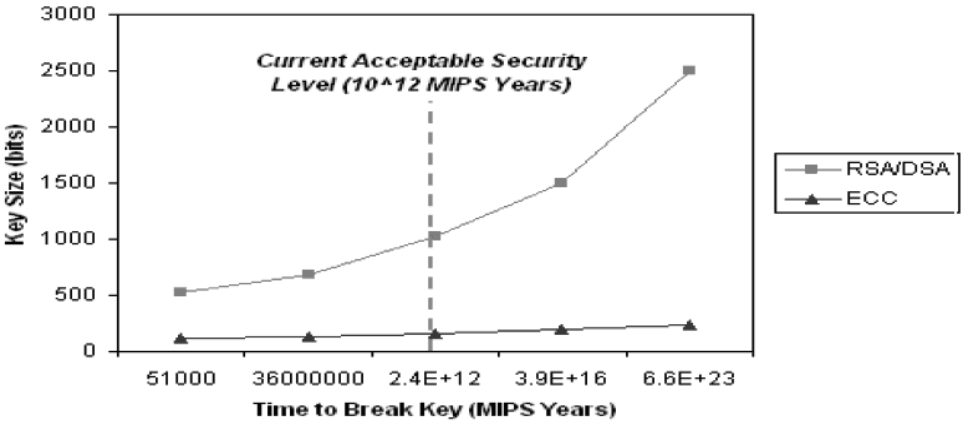

Figure 2. Comparison of securities level of ECC and RSA & DSA

Table 2. Performance comparison between ECC and RSA

RSA		ECC (bits)		Time Ratio	CPU
Key Length (bits)	**Time (Seconds)**	**Key Length (bits)**	**Time (Seconds)**		
1024	105	160	4.6	20 : 1	8051 – 8 bit micro controller (14.75 MHz)
1024	22	160	1.62	14 : 1	Atmel AVR - 8-bit micro controller (4 Mhz)
2048		224		50 : 1 ~ 100 : 1	Future key sizes

Solution for issue 2. The first case is that when a new user join a private group or a personal group in HangOut, the group key will be sent to that user. And this key is shared for all group members, they will use this group key to encrypt all personal messages before sending it to the others in this group, and the others will use the group key again to decrypt and read the message. How to know that the shared group key is the real key of that group? Since an adversary or an intermediate can listen to the joining request, then intervene, drop the real group key and send the fake key of the adversary to the new user, and repeat the same action to the other new users, then all messages can read by the adversary. The second case is that the public key of the location is published to all mobile clients actually is not guaranteed that it is the real public key. So we need one server do the task which authenticating the key group is the real key of that group.

The group key is used in the personal group and private group. In HangOutPlus, the shared group key is not stored in the mobile client. There is more one server as the CA (Certificate Authority) server which generates and stores the group key. When a new user would like to join a group, they will send a registration request to the CA server to receive the group key by clicking the "sign up" link on the mobile device screen, and the shared group key is exchanged by using Diffie-Hellman exchange key algorithm [30] between the CA server and the new user, then user has the CA key. The CA server will encrypt the group key and send to the user. The user will use the CA key to decrypt that message to get the group key. After a random period of time *t'*, the CA server will resend the new group key, resend to all legal users in that group. This ensures that an adversary cannot use the old group key to communicate with the other group legal users.

The public key of the location server also will authenticated through the CA server. The public key is also re-generated randomly after a random period of time by CA server, then resend to all legal users in the current system.

Solution for issue 3. In order to make it difficult for adversaries to find out the links between mobile devices and messages. We will generating dummy messages to fix this issue. A dummy message is a fake message introduced in a mix network in order to make it more difficult for an attacker to deploy passive and active attacks. The purpose of using dummy message is that when database server receives an update request.

We also need define how many dummy messages are generated in the system and how long is the period of time for generating dummy messages?

In this article, the database server will generated dummy messages in a p by using geometrical distribution [31]. The algorithm based on geometrical distribution as follows:

```
Initiate:
Create a random real number p belonging to the domain [0, 1]
  N=0
Repeat:
    N = N+1
    F(x) = 1 | p^N
    Create a random real number u belonging to [0, 1]
    When (u>f(x))
           Return N
```

(*N-f*) is the number of dummy messages generated by servers with f as the real messages going through the system.

And servers will update entries after a period of time t_{up} which is randomly auto-generated and also changes after a random period of time t_t. After a time t_{up} when database server receives the update request, if the number of update request is n which is not equal to n_o (the number of request messages servers will wait to receive), it will update entries as request, beside it will write a number (n_o-n) messages chosen randomly in the database to be overwritten by itself. This ensures there are always constant number of entries in the database updated.

Solution for issue 4. HangOutPlus uses the searchable encryption algorithm for the database in [32] proposed by Thuc D. Nguyen et al. This algorithm is a kind of search encryption, a way which enables a recipient to give a un-trust server the ability to test whether W is a keyword in a large message M but server should learn nothing else about the keyword W and the message M.

In this scheme, a sender B who want to send a secrete message to a recipient A via a un-trust server. The scheme is briefly described as the following:

- The sender B encrypts his message using a standard public key system. He then appends to the resulting cipher-text a public key encryption with keyword search (PPEKS) for each keyword. B send a message M with keywords W1,…,Wp to C: $E_{Apub}(M)||PPEKS(A_{pub},W1)||…||PPEKS(A_{pub},Wp)$, where A_{pub} is A's public key and E a encryption function.
- The recipient A gives the 3rd party C a certain trapdoor T_W that enables C to test whether one of the keywords associated with the message is equal to the work W of A's choice: given $PPEKS(A_{pub},W')$ and T_W, C can test if W=W'

So if we decide to use this algorithm for HangOutPlus, we will not use ECC algorithm in the solution for issue 1 [31]. This algorithm runs so fast, since the main operator in this algorithm is XOR operator between zero bits and one bits. Moreover, it is totally suitable to mobile device's limited resource and performance.

With this scheme, even database server is compromised by an adversary, the data is still secure. In HangOutPlus, keywords B sends with B's messages are group number and member number as in HangOut, but they are encrypted. The database server will store those encrypted records tailed the encrypted keywords. And the algorithm helps the database find out necessary records effectively without knowing anything about group number and member number of mobile devices.

5. CONCLUSION AND FUTURE WORKS

In this article, we present the overview of HangOut system which is a privacy preserving location based social networking service proposed by Annavaram, Jacobson and Shen in [1]. We also analysis the current issues in HangOut system. In order to fix these issues, we strongly proposed some solutions. After HangOut is modified with our methods, it is called HangOutPlus. HangOutPlus can keep the real-time attribute by using ECC or the assymetric based on group thoery which having the same security level with AES, instead of using RSA. HangOutPlus is also more secure protocol since it has the effective key exchange and management protocol, and it uses the key searchable protocol instead of decrypting the message and searching the records in the database in order to overwrite the old location record with the updated information.

However, HangOut as well as HangOutPlus has one remaining issue that is using SMS interface to share the private key with new members when they join the personal group or the private group. Because adversaries can listen on SMS communications and can read the SMS message which is also the sensitive private key. So, the future works is fixing this issues and deploying HangOutPlus in real life.

Acknowledgments. We would like to express my thanks to Dr. Nguyen Dinh Thuc and the security group at Faculty of Information Technology, in University of Science, VNU – HCMC for supporting us.

REFERENCES

1. Murali Annavaram, Quinn Jacobson, John P. Shen - HangOut: A Privacy Preserving Location Based Social Networking Service. In: the 11th international symposium on Modeling, analysis and simulation of wireless and mobile systems, ISBN:978-1-60558-235-1, pp. 229 -- 238, ACM, New York, USA, 2008.
2. 2010 Social Networking Websites Review Comparisons, http://social-networking-websites-review.toptenreviews.com/
3. Article "A Collection of Social Network Stats for 2010, http://www.web-strategist.com/blog/2010/01/19/a-collection-of-social-network-stats-for-2010/
4. Article "Social Networks Overtake Search Engines In UK" in Social Network Watch, http://www.socialnetworkingwatch.com/2010/06/social-networks-overtake-search-engines-in-uk.html
5. Gross R., Acquisti A. - Information Revelation and Privacy in Online Social Networks, Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society (WPES), pp. 71-80, New York, 2005.

6. Viegas F. - Bloggers' Expectations of Privacy and Accountability: An Initial Survey, Journal of Computer-Mediated Communication **10** (3) (2005).
7. Lam I., Chen K., and Chen L. - Involuntary Information Leakage in Social Network Services, Proceedings of 2008 International Workshop on Security, Kanagawa, Japan, 2008, pp.167-183.
8. Calvo-Armengol A., Zenou Y. - Social Networks and Crime Decisions: The Role of Social Structure in Facilitating Delinquent Behavior, International Economic Review **45** (3) (2004) 939-958.
9. E. Michael Maximilien, Tyrone Grandison, Tony Sun, Dwayne Richardson, Sherry Guo, Kun Liu - Privacy-as-a-Service: Models, Algorithms, and Results on the Facebook Platform, Workshop Program - W2SP 2009: Web 2.0 Security and Privacy, 2009.
10. Charu C. Aggarwal, Philip S. Yu - Privacy-Preserving Data Mining: Models And Algorithms, Chapter 2, Kluwer Academic Publishers.
11. Tran Hong Ngoc, Isao Echizen, Komei Kamiyama, Hiroshi Yoshiura - New Approach to Quantification of Privacy on Social Network Sites, the 24th IEEE Advanced Information Networking and Applications International Conference - AINA, Perth, Australia, 2010.
12. Hoh B. and M. Gruteser - Protecting Location Privacy through Path Confusion, in First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM 2005), 2005, IEEE Computer Society: Athens, Greece. pp. 194-205.
13. Duckham M. and L. Kulik - Simulation of Obfuscation and Negotiation for Location Privacy, in Spatial Information Theory, International Conference, COSIT 2005, 2005, Springer: Ellicottville, NY, USA. pp. 31-48.
14. Gruteser M. and D. Grunwald - Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking, in First ACM/USENIX International Conference on Mobile Systems, Applications, and Services (MobiSys 2003), 2003, ACM Press: San Francisco, CA USA. pp. 31-42.
15. Hoh B., et al. - Preserving Privacy in GPS Traces via Uncertainty-Aware Path Cloaking, in 14th ACM Conference on Computer and Communication Security (ACM CCS 2007). 2007: Alexandria, VA USA.
16. Hashem T. and L. Kulik - Safeguarding Location Privacy in Wireless Ad-Hoc Networks, in 9th International Conference on Ubiquitous Computing (UbiComp 2007). 2007: Innsbruck, Austria. pp. 372-390.
17. John Krumm - A survey of computational location privacy, Personal and Ubiquitous Computing **13** (6) (2008).
18. Sweeney L. - Achieving k-Anonymity Privacy Protection Using Generalization and Suppression. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems **10** (5) (2002) 571-588.
19. N. Li and T. Li - t-closeness: Privacy beyond k-anonymity and l-diversity, In: Proceedings of the 23nd International Conference on Data Engineering (ICDE '07), Istanbul, Turkey, Apr. 16-20 2007.

20. A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam - 1-diversity: Privacy beyond k-anonymity, In: Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006), Atlanta Georgia. USA, 2006.

21. L. Capra, W. Emmerich, and C. Mascolo - A microeconomic approach to conflict resolution in mobile computing. In Proceedings of the 10th symposium onFoundations of software engineering, pages 31-40, 2002.

22. J. Tang, V. Terziyan and J. Veijalainen - Distributed PIN verification scheme for improving security of mobile devices. In Mobile Networks and Applications **8** (2) (2003) 159-175.

23. B. Hoh, M. Gruteser, M. Annavaram, Q. Jacobson, R. Herring, J. Ban, D. Work, J. Herrera, and A. Bayen - Virtual trip lines for distributed privacy-preserving traffic monitoring, To Appear in Proceedings of the 6th international conference on Mobile systems, applications and services, June, 2008.

24. Researchers Test GPS-Cell Phone Navigation In South Bay, NBC News, February 2008, http://www.nbc11.com/news/15255056/detail.html.

25. C. A. Ardagna, A. Stavrou, S. Jajodia, P. Samarati, R. Martin - A multi-path approach for k-anonymity in mobile hybrid networks. In Bettini, S. Jajodia, P. Samarati, and S. Wang, (eds.), Privacy in Location Based Applications, Springer, 2009.

26. Bhuvan Bamba, Ling Liu, Peter Pesti, Ting Wang - Supporting Anonymous Location Queries in Mobile Environments with PrivacyGrid, the International World Wide Web Conference Committee (IW3C2), Beijing, China, 2008.

27. C. A. Ardagna, S. Jajodia, P. Samarati, A. Stavrou - Privacy Preservation over Untrusted Mobile Networks, In: Privacy in Location-Based Applications , Vol. 5599, Springer, 2009, pp. 84-105.

28. Bill Burr Manager - Security Technology Group NIST, NIST cryptographic Standards Status Report, April 4, 2006.

29. Sheueling Chang Shantz - Next-GenerationInternet Security, Disttinguished Engineer, Sun Laboratories, 2004.

    http://labs.oracle.com/sunlabsday/docs.2004/talks/2.03_Chang.pdf

30. David A. Carts - A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols, SANS Institute, 5 November 2001.

31. Geometric Distribution, MathWolf World,

    http://mathworld.wolfram.com/GeometricDistribution.html.

32. Thuc D. Nguyen - Peeter Laud, Van H. Dang, a public key encryption with keyword search scheme based on pseudo-inverse matrix (Draft), 2010.

Address:

Faculty of Information Technology,

University of Science, Vietnam National University.

Received June 16, 2010