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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

Sustaining Isolation Security for Preserving User's Seclusion in Web Search Engines

P.Markandeyulu[1], K.Narayana[2]

Student, Dept of Computer Science Engineering, Seshachala Institute of Technology, Chittoor (D), India[1]

Associate Professor & HOD, Dept of Computer Science Engineering, Seshachala Institute of Technology, Chittoor (D), India[2]

ABSTRACT: We apply a PWS support called UPS that can adaptively simplify profiles by queries as regarding user specific privacy necessities. Our runtime simplification aims at conspicuous a equilibrium among two analytical metrics that estimate the service of personalization and the isolation danger of revealing the general outline. We present two greedy algorithms, specifically Greedy DP and Greedy IL, for runtime simplification. We also present an online prediction machinery for deciding whether personalizing a query is useful. Wide-ranging experiments demonstrate the effectiveness of our framework. The experimental results also expose that Greedy IL significantly outperforms Greedy DP in terms of efficiency. In this paper we present a novel procedure specifically calculated to protect the users' privacy in front of web search profiling. Our system provides a distorted user profile to the web search engine. We offer implementation details and computational and communication results that show that the proposed protocol improves the existing solutions in terms of query delay. Our scheme provides an reasonable overhead while contribution seclusion benefits to the users.

KEY WORDS: PWS framework, UPS, Greedy DP, Greedy IL, Profiling.

I. INTRODUCTION

The solutions to PWS can normally be categorized into two types, specifically click-log-based methods and profile-based ones. The click-log based methods are basic, they simply impose bias to clicked pages in the user's query history. Though this strategy has been demonstrated to perform consistently and significantly well, it can only occupation on frequent queries from the same user, which is a strong constraint confining its applicability. In distinction, profile-based methods recover the discover knowledge with difficult user-interest models generated from user profiling techniques. Profile-based methods can be potentially efficient for approximately all sorts of queries, but are reported to be unstable under some conditions. Web search engines (e.g. Google, Yahoo, Microsoft Live Search, etc.) are extensively used to do this hard job for us. The 84% of the Internet users have used a web search engine at least once. For the 32%, web search engines are an necessary tool to address their everyday duties [1]. Among the different search engines, Google is the most used in the US with a 43.7% of the total amount of searches performed in 2006 [2]. Google improves its performance (it gives personalized search results) by storing a record of visited sites and past searches submitted by each user [3] (Web History).

Those searches can expose a lot of information from individual users or the institutions they work for. For example, let us imagine an employee of a certain company A. This employee uses Google to obtain information about a certain knowledge. If a company B, which is a direct participant of A, knows this circumstances, it can infer that this technology will be used in the new products offered by A. This knowledge gives to B an important advantage over A. Another example of this situation occurs when a person is applying for a certain job. In this case, if the employer knows that the applicant has been looking for information regarding a confident disease; she can use this knowledge to choose one more person for the job. In both examples, the aggressor (the entity who gets some advantage over the other) benefits from the lack of a privacy-preserving method among the user and the web search engine.

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

II. CRYPTOGRAPHIC BUILDING BLOCS

In this segment, we summarize the cryptographic tools our protocol is based on.

n-out-of-n entrance ElGamal encryption

In cryptographic multi-party protocols, some operations must be computed equally by dissimilar users. In an n-out-of-n entrance ElGamal encryption [14], n users share a public key y and the equivalent unknown private key a is separated into n shares a α_i. Using this protocol, a positive communication m can be encrypted using the public key y and the decryption can be performed only if all n users collaborate in the decryption process. Key generation, encryption and decryption procedure are next described.

Key in production

First, a large random prime p is generated, where p = 2q + 1 and q is a most important number too. Also, a generator g of the multiplicative group $\mathbb{Z}_q^*$ is chosen. Then, each user generates a random private key $\alpha_i \in \mathbb{Z}_q^*$ and publishes $y_i = g^{\alpha_i}$. The ordinary public key is computed as $y = \prod_{i=1}^{n} y_i = g^{\alpha}$, where $\alpha = \alpha_1 + \cdots + \alpha_n$.

Grouping set up

While a user U_i needs to present a doubt to the web search engine, she sends a message to the central node C requesting to be included in a group. The middle node C receives all user requests. Once it has n requests, it creates a new user group $\{U_1, \ldots, U_n\}$ and notifies the n users that they belong to the same group. After this first step, users in this group can establish a message channel between them. In this way, each user can send messages to the rest of the group directly. The essential node is no longer needed.

Group key creation

The UUP procedure uses a group key which is generated using a fair entrance encryption scheme. All users follow these steps:

(1) Users $\{U_1, \ldots, U_n\}$ agree on a large prime p where $p = 2q + 1$ and q is prime too. Next, they pick an element $g \in Z_p^*$ of order q.

(2) Users $\{U_1, \ldots, U_n\}$ produce an ElGamal single public key y using the n-out-of-n threshold ElGamal encryption.

Message decryption

Given a significance encrypted with a public key $y, E_y(m, r) = (c1, c2)$, user U_i can decrypt that value as follows: Each user $[j \neq i]$ publishes $c1^{\alpha_i}$. Then, U_i can recover message m in $m = \frac{c2}{c1^{\alpha_i}\left(\prod_{j \neq i} c1^{\alpha_j}\right)}$ This decryption can be verified by each participant by performing a proof of equality of discrete logarithms [16].

Defense analysis

Our proposal has been designed to protect the privacy of the users when they submit queries to a web search engine. According to that, a successful attacker would be able to link a certain query to the user who has generated it. We believe that the computational power of an attacker does not allow him to break current computationally protected cryptosystems. our protocol assumes that the users follow the proposed protocol correctly and that there are no collusions between entities. The attacker can be any entity (one of the three entities of the protocol or an external one). However, external attackers can get, at most the same information that an internal entity. For that reason, we peorm our analysis assuming that the attacker is an interior entity.

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

III. ACHIEVEMENT INFORMATION AND TENTATIVE RESULTS

The protocol prevents a search engine from obtaining a reliable profile of a certain user, i.e. it brings a higher degree of retreat to the users. From the firmly industrial point of view, the cost of achieving this privacy degree can be measured in terms of query delay. The planned protocol requires some cryptographic operations and network communications that increase this delay. We have implemented our protocol and we next present some results regarding its performance in a practical scenario. These results prove that our suggestion introduces a delay which can be assumed by the user.

Accomplishment and configuration information

The future system requires two mechanisms: the essential node and the client submission. These two mechanisms have been implemented using the Java programming language [18]. This allows application portability. The central node C is a process (daemon) that listens to client requests in a fixed TCP port. Following getting n requirements, C creates a new group and sends a message with the users' IP address and the number of the port to be used. In order to improve the protocol's presentation, this message also contains the large prime p, and the $g \in Z_{p}^{*}$ This reduces the number of messages that must be transmitted. The configuration of the central node includes the number n of users needed to form a group, the port number, and the length of the large prime p. The client request is a java applet which is accessed by an html web page (see Fig. 1) that allows users to make a search in a transparent manner. The client interface displays a form field (similar to the one that can be launch in a classic web search engine) where the user must type her query. The search process is started once the search button is pressed. The text is sent from the form field to the applet using a Javascript code. The applet runs the proposed UUP protocol establishing connections with all the group members and with the web search engine. Finally, it shows the result to the user. All transportation between entities are performed using TCP connections. Messages are implemented using the XML format.

Testing methodology

The UUP protocol can be configured with two different parameters: the group size n and the key length l used by the cryptographic operations. Obviously, the larger n is, the better the algorithm hides the real profile of each user. However, when a user wants to submit query, she has to wait until another n -1 users want to do the same. Thus, when n increases, the waiting time needed to create a group also increases. In the same way, a larger group implies more messages between the group members and therefore a higher delay due to the announcement. The key length l refers to the size (in bits) of the ElGamal cryptosystem used in the UUP protocol. A short key is considered not secure, hence it is required a minimum key length. On the other hand, the time consumed in the cryptographic operations (key generation, encryption, decryption and re-masking) of the UUP protocol is directly related to the key size used. The UUP protocol should use a large n and l to hide the real users' profiles and to offer enough security. At the same time, n and l should be short in order to introduce a low delay in the response. According to that, there is a trade-off for both parameters. In order to tune our system and to find the values of n and l that offer a reasonable combination of privacy, security and usability, we have used two testing environments. These are a controlled environment and an open surroundings The controlled environment is a Local Area Network (LAN). The time obtained by the UUP protocol in this environment is not affected by external factors. Thus, it allow us to evaluate the protocol behaviour in optimal conditions and perform a tuning analysis of the values n and l. In this surroundings, for each key length, we have run the protocol with different group sizes to see how both parameters affect the query delay. The group size n and the key length. The results obtained in the tuning analysis within the controlled environment show the maximum n and l values which can be used to get a low query delay. In this environment, the protocol is evaluated in optimal conditions. Therefore, any combination of n and l that results in a high query delay is very likely to be unaffordable in the open environment (Internet). According to that, the controlled environment provides the best configuration. Then, these configurations are tested in the open environment in order to get the real delay introduced by our proposal.

Time measures

Sections provide aggregated results in milliseconds that show the delay introduced by our scheme when submitting a single query. In addition to that, offer a detailed analysis of the time spent on each protocol step. This is done to show which steps are the most sensitive ones. Below, there is a description of each time interval:

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

Fig. 1. Client interface.

Table 1
Controlled environment: equipment.

Computers	CPU	Intel Core 2 CPU 6320 at 1.86 GHz
	RAM	2 GBytes
	O.S.	Microsoft Windows Server 2003
	Java version	Java 6 Update 5
	Ethernet	100 Mbits
Network	Switch 100 Mbits	

Controlled testing environment
In the controlled environment, there is no traffic from external sources. Therefore, the resulting delay can only be attributed to the proposed protocol. we justify the selection of parameters n and l which has been used in our tests. we provide a detailed analysis of the time spent on each protocol step. Time delay experienced by our protocol in the controlled environment.

IV. OPEN SURROUNDINGS

The results obtained in the controlled environment are not realistic. However, they are useful to understand the effects of the group size and the key length on the system. Presents the parameters used in the open environment. The time intervals obtained in the open environment.

Bound selection
According to the time obtained in the controlled environment we used n =3 and l =1024 in the open environment. On one hand a key length below 1024 is not considered safe [19], on the other hand, a key length larger than 1024 increases the computing time. Thus, a length of 1024 bits was selected as the best trade-off between both concepts. Regarding the group size, we performed our tests with n = 3 because in the controlled environment the query delay was larger than 2.5 s for better groups.

Equipped considerations
In previous sections, we have discussed the technical details of the proposed system regarding implementation issues and equipment necessities. Besides, we have also analyzed the computation and communication costs. However, there are other operational considerations that should be taken into account when deploying this scheme in a real condition. First of all, from the strictly technical point of view, the proposed scheme provides a certain level of privacy in front of the web search engine at the cost of introducing an affordable delay in the query process. However, from the user point of view, another cost is added in the sense that users are submitting queries from other users. Obviously, a certain user may not be satisfied with the content of other users' queries and may not desire to submit them to the web search engine. This problem can be solved using a filter that discards queries depending on their content. This filter must be carefully designed in order to not require the final user to perform an extreme amount of work. How to implement this is outside the scope of this paper and requires further research. Two possible directions to address this issue would be: (i) allow the users to select categories of words to be discarded. These categories would be based on the ones defined

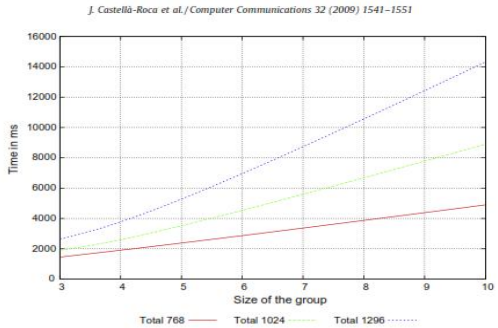
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

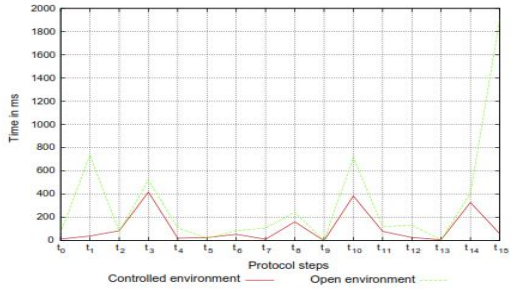
(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

by the Open Directory Project (ODP) [20]; and (ii) use ontologies to process the content of the queries. Refs. [21,22] are examples of the work done in this field.



Average time required to submit a query with our system. On the other hand, the proposed scheme is based on a central server that connects different users who want to submit a query. Note that the workload of the central server is really low since it only offers to users contact details about other users. The rest of the steps needed to form the group are entirely performed by the users. According to that, it is possible to assume that some central nodes can be voluntary working to improve the privacy of the users. In fact, the Tor network, which has a similar privacy preserving objective, is based on the use of voluntary nodes (more than 1800 [23]). Besides, the workload of a Tor node is higher than the workload of the central node used in our scheme. Finally, another operational consideration arises regarding the number of users needed to form a group and the time each user has to wait until that number is available. The main objective of our proposal is not to offer anonymity to the users who submit queries to a web search engine. Instead of that, it provides non-accurate user profiles to the web search engine. According to that, the system is still valid if the users submit some of their own queries. This can be implemented in the client request using a timeout. If, after this timeout, the group has not been formed, the user can send her query to the web search engine directly. The correctness of this measure can be controlled by keeping track of the total number of queries that have been submit directly. This number must be kept below a predefined threshold in order to provide a useless profile to the web search engine. Defining the correct value for this threshold requires a detailed study which is not the purpose of this paper. Nevertheless, the work presented in [24] can give us an approximation of this value. In [24], the system submits s fake queries for each legitimate query. In order to define the privacy level which is achieved, the system uses one vector to store all the fake queries that have been submitted and another vector to store the legitimate ones. The comparison between both vectors defines the privacy level achieved. The results provided by Kuflik et al. [24] show a fairly good privacy level when $2 \leqslant \tau \leqslant 4$.



Partial times with a key length of l = 1024 bits and n = 3 in the restricted and open environments.

V. CONCLUSIONS

Proposed system accessible a client-side isolation defense framework called UPS for modified web search. UPS could potentially be adopted by any PWS that captures user profiles in a hierarchical classification. The framework allowed users to specify tailored privacy necessities via the hierarchical profiles. In addition, UPS also performed online simplification on user profiles to protect the individual privacy without compromising the search quality. We proposed

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 1, January 2015

two greedy algorithms, namely GreedyDP and GreedyIL, for the online simplification. Our investigational results exposed that UPS could achieve quality search results while preserving user's customized privacy requirements. The results also confirmed the effectiveness and efficiency of our solution. In this paper, a novel protocol for protecting the users' privacy when dealing with a web search engine has been proposed. Our system does not need any change in the server side and, moreover, the server is not required to collaborate with the user. The proposed protocol has been implemented to prove its functionality. Statistical results of the protocol's performance show that the presented scheme improves previous proposals. In calculation to that, these results also prove that this protocol can be practical in actual scenarios.

REFERENCES

[1] D. Fallows, Search engine users: internet searchers are confident, satisfied and trusting, but they are also unaware and naïve, Pew/Internet & American Life Project (2005).
[2] D. Sullivan, comScore Media Metrix Search Engine Ratings, comScore, 2006. Available from: <http://searchenginewatch.com>.
[3] Google History, 2009. Available from: <http://www.google.com/history>.
[4] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, Private information retrieval, in: IEEE Symposium on Foundations of Computer Science – FOCS, 1995, pp. 41–50.
[5] B. Chor, O. Goldreich, E. Kushilevitz, M. Sudan, Private information retrieval, Journal of the ACM 45 (1998) 965–981.
[6] E. Kushilevitz, R. Ostrovsky, Replication is not needed: single database, computationally-private information retrieval, in: Proceedings of the 38th Annual IEEE Symposium on Foundations of Computer Science, 1997, pp. 364–373.
[7] R. Ostrovsky, W.E. Skeith III, A survey of single-database PIR: techniques and applications, Lecture Notes in Computer Science 4450 (2007) 393–411.
[8] B. Chor, N. Gilboa, M. Naor, Private information retrieval by keywords, Technical Report TR CS0917, Department of Computer Science, Technion, 1997.
[9] Tor Project, 2009. Available from: <http://www.torproject.org>.
[10] F. Saint-Jean, A. Johnson, D. Boneh, J. Feigenbaum, Private web search, in: Proceedings of the 2007 ACM workshop on Privacy in electronic society – WPES'07, 2007, pp. 84–90.
[11] X. Shen, B. Tan, C.X. Zhai, Privacy protection in personalized search, ACM SIGIR Forum 41 (1) (2007) 4–17.
[12] M. Barbaro, T. Zeller, A face is exposed for AOL searcher No. 4417749, New York Times, August 2006.
[13] Y. Xu, B. Zhang, Z. Chen, K. Wang, Privacy-enhancing personalized web search, in: International World Wide Web Conference, 2007, pp. 591–600.
[14] Y. Desmedt, Y. Frankel, Threshold cryptosystems, Advances in Cryptology – CRYPTO'89, Lecture Notes in Computer Science 335 (1990) 307–315.
[15] T. ElGamal, A public-key cryptosystem and a signature scheme based on discrete logarithms, IEEE Transactions on Information Theory 31 (1985) 469–472.
[16] D. Chaum, T. Pedersen, Wallet databases with observers, Advances in Cryptology – CRYPTO'92, Lecture Notes in Computer Science 740 (1992) 89–105.
[17] M. Abe, Mix-networks on permutation networks, Advances in Cryptology – Asiacrypt'99, Lecture Notes in Computer Science 1716 (1999) 258–273.
[18] Sun Microsystems, JAVA Programming language, 2008. Available from: <http://java.sun.com>.
[19] Recommendation for Key Management, Special Publication 800–57 Part 1, NIST, 2007.
[20] Open Directory Project, 2009. Available from: <http://www.dmoz.org/>.
[21] D. Brewer, S. Thirumalai, K. Gomadamk, K. Li, Towards an ontology driven spam filter, in: Proceedings of the 22nd International Conference on Data Engineering Workshops, 2006.
[22] S. Youn, D. McLeod, Efficient spam email filtering using adaptive ontology, in: Proceedings of the International Conference on Information Technology, 2007, pp. 249–254.
[23] Tor Node Status, 2009. Available from: <https://torstat.xenobite.eu/>.
[24] T. Kuflik, B. Shapira, Y. Elovici, A. Maschiach, Privacy preservation improvement by learning optimal profile generation rate, Lecture Notes in Computer Science 2702 (2003) 168–177.
[25] D. Rebollo-Monedero, J. Forné, L. Subirats, A. Solanas, A. Marti nez-Ballesté, A collaborative protocol for private retrieval of location-based information, Proceedings of the IADIS International Conference on e-Society, Barcelona, Spain, February 2009.