Hindawi
Security and Communication Networks
Volume 2022, Article ID 2109024, 13 pages
https://doi.org/10.1155/2022/2109024

Research Article

Profile Aware ObScure Logging (PaOSLo): A Web Search Privacy-Preserving Protocol to Mitigate Digital Traces

Mohib Ullah,[1] **Rafi Ullah Khan**,[1] **Irfan Ullah Khan**,[2] **Nida Aslam**,[3] **Sumayh S. Aljameel**,[2] **Muhammad Inam Ul Haq**,[4] **and Muhammad Arshad Islam**[5]

[1]*Institute of Computer Science and Information Technology, The University of Agriculture, Peshawar, Pakistan*
[2]*Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam 31441, Saudi Arabia*
[3]*SAUDI ARAMCO Cybersecurity Chair, Department of Computer Science, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam 31441, Saudi Arabia*
[4]*Department of Computer Science and Bioinformatics, Khushal Khan Khattak University Karak, Pakistan*
[5]*National University of Computer and Emerging Sciences, Islamabad, Pakistan*

Correspondence should be addressed to Mohib Ullah; mrmohibkhan@gmail.com

Received 27 August 2021; Accepted 23 November 2021; Published 3 February 2022

Academic Editor: Farhan Ullah

Web search querying is an inevitable activity of any Internet user. The web search engine (WSE) is the easiest way to search and retrieve data from the Internet. The WSE stores the user's search queries to retrieve the personalized search result in a form of query log. A user often leaves digital traces and sensitive information in the query log. WSE is known to sell the query log to a third party to generate revenue. However, the release of the query log can compromise the security and privacy of a user. In this work, we propose a Profile Aware ObScure Logging (PaOSLo) Web search privacy-preserving protocol that mitigates the digital traces a user leaves in Web searching. PaOSLo systematically groups users based on profile similarity. The primary objective of this work is to evaluate the impact of the systematic group compared to random grouping. We first computed the similarity between the users' profiles and then clustered them using the K-mean algorithm to group the users systematically. Unlikability and indistinguishability are the two dimensions in which we have measured the privacy of a user. To compute the impact of systematic grouping on a user's privacy, we have experimented with and compared the performance of PaOSLo with modern distributed protocols like OSLo and UUP(e). Results show that, at the top degree of the ODP hierarchy, PaOSLo preserved 10% and 3% better profile privacy than the modern distributed protocols mentioned above. In addition, the PaOSLo has less profile exposure for any group size and at each degree of the ODP hierarchy.

1. Introduction

Web search engines (WSEs) like Google, Ask, Bing, AOL, Baidu, and others provide the easiest way to search and retrieve information from the Web. The WSE stores the users' submitted queries in a query log. The WSE regularly builds and updates a user profile from the query log to provide personalized results [1]. The WSE generates revenue by analyzing the query log coupled with a user profile to provide relevant advertisements [2, 3]. Research shows that 35% of products people buy on Amazon and 75% of videos they watch on Netflix result from personalized recommendations [4]. The user query log often contains sensitive information, and the release of such information poses a risk to user's security and privacy [2, 5]. In today's Internet life, preserving web search privacy is the real perturb of a user. Existing techniques hide the identity through unlinkability and obfuscate the profile through indistinguishability to succeed the Web search privacy of a person [2, 6]. Internet users often use proxy services (like scoogle.com, anonymizer.com, and others) and TOR (the onion routing) network to attain unlinkability [7], whereas users utilise TrackMeNot [7], GooPIR, and DisPA [8] to achieve indistinguishability by sending fictitious but real

queries to obfuscate the profile maintained by the WSE [2]. However, the WSE can recognise TOR users' queries from the cookies and application layer. Similarly, the unlinkability achieved through proxy services can be more precarious to users as the privacy policies of the proxy servers are not as regulated as the WSE.

The distributed privacy-preserving protocol is another approach that provides both unlinkability and indistinguishability. It works on the cooperation of multiple users. The distributed protocols create a group of "n" users who need to query the WSE secretly. Group users send each other queries to WSE and broadcast the results received in the group. A user achieves unlinkability in distributed protocols as a user's query is forwarded by another group user. Similarly, a user attains indistinguishability by forwarding queries of the other group of users. Such measures obscure the users' profiles with group member queries. Hence, the user's profile kept at WSE includes queries of other group users.

The grouping of users is considered a primary step in distributed protocols. In the existing approaches, a Core Server (CS) accepts a connection request from individuals who want to perform a Web search secretly. Upon receiving "n" number of connection requests, the CS creates a group of users on a first-come first-serve basis, also called random grouping. The major shortcoming of a random grouping is that a user may be grouped with those users who have similar interests, as there is no prior information about the users' preferences [9]. Consider a situation where a user Mr. X has a medical-related query. He is in a group with other users having a similar interest; in such a case, although Mr. X forwards a query of another user, his profile will not be significantly obfuscated. The profile maintained by the WSE for Mr. X will contain the same type of categories. Profile obfuscation is the fundamental objective of web search privacy; however, a user can be grouped with similar interests with an existing randomized grouping. Such grouping will not expressively obfuscate the profiles of the users. Therefore, a mechanism is required to systematically group users based on their interests.

This work proposes a novel distributed privacy-preserving protocol, PaOSLo (Profile Aware ObScure Logging), to significantly obfuscate a user's profile by systematically grouping users instead of random grouping. The PaOSLo executes in two steps: (i) cluster the user according to their profile similarity. (ii) The CS creates a group of users from each distinct cluster instead of random grouping. PaOSLo aims to achieve the following objectives:

(1) To notably obfuscate the profile of a user by creating a profile-aware grouping mechanism

(2) An experiment is performed to estimate the extent of profile obfuscation by utilizing profile-aware grouping versus randomized grouping

A K-mean algorithm is employed to set users into three clusters, four clusters, and five clusters. The similarity between the users' profiles is computed using the cosine similarity metric.

The rest of this article is represented as existing work which is discussed in Section 2. PaOSLo and its execution are explained in Section 3. Section 4 describes the dataset and simulation details. Section 5 presents PaOSLo privacy evaluation. Section 6 details results and discussion. The last section of the article demonstrates the conclusion and future work.

2. Existing Work

As discussed above, unlinkability and indistinguishability are the advantages of distributed protocols compared to other privacy-preserving schemes. Crowds was the first distributed protocol proposed to protect web search privacy [10]. However, it was vulnerable to an active and passive adversary [11]. User private information retrieval (UPIR) presented by Domingo-Ferrer et al. used memory spaces as drop boxes to achieve indistinguishability [12]. However, the query remained visible to the users associated with the same memory location, and UPIR was vulnerable to intersection attacks. Swanson and Stinson extended the model of UPIR and added standard terminologies in the field of privacy [13]. They calculated the probabilistic advantage to a user connected with the same dropbox in linking a query with the originator. Swanson and Stinson again extended the UPIR, and they figured the privacy relative to peer users when they make a coalition to link a query with the originator [14]. However, this extended technique suffered the same shortcomings. First, the search query remained visible to group users; second, group user collaboration could reveal the user's identity. Castella Roca et al. proposed a Useless User Profile (UUP) protocol. It employed a central server (CS) to create a group of "n" users; the queries were shuffled among the users before forwarding to the WSE [15]. UUP results were broadcast in clear text, letting everyone know the intention of other group users. Romero-Tris et al. integrated the ElGamal key encipherment to attain concealment and the optimized Benes network to shuffle the queries in UUP to achieve privacy in the existence of an untrusted partner [16]. However, the extended UUP (UUP (e)) was still unprotected as the researchers compromised a user's privacy through machine learning attacks [5]. Ullah et al. presented ObScure logging (OSLo) to lessen the deficiencies of previous techniques and preserve a user's privacy. The central server (CS) in OSLo exercised a single dynamic group of "n" random users [11]. OSLo encrypted both the query and results to achieve confidentiality and shuffled the queries with the flip of a coin to attain unlinkability. However, there was no system for group creation; any user can be grouped with another user. Due to random grouping, the chances of being grouped with users having the same interests were higher. Domingo-Ferrer et al., to support social welfare, introduced a novel concept of a self-enforcing protocol called coutile [17]. To send a query, a user must check the query if the query would obfuscate the user's profile or expose the profile. In the former case, the user sends the search keyword directly to WSE, while in the latter case, the user asks the peer member to deliver the query on their behalf. When a user asks a peer for query forwarding, he/she follows the same steps. The peer only forwards if the query obscures the profile and denies the

request otherwise. Delay is a primary issue in coutile protocol, a user waits for a longer time to get a query answered, but the user's request gets denied for not obfuscating the profile of a peer user. Such a situation can cause a significant delay in getting results for the query. Authors have presented MG-OSLo to obfuscate the profile of users by employing multiple groups [2]. MG-OSLo achieved both local privacy and profile privacy through unlinkability and indistinguishability. However, the same random grouping is used in MG-OSLo, and the chance of being grouped with users having similar interests remains the same. Kaaniche et al. proposed a decentralized solution CoWSA that empowers end-users to have control over personal data, mitigates single-point failure, ensures the security of the queries, and provides anonymity to a user [1]. User, client, WSE, third parties (TP), and trusted authorities are the five entities of CoWSA. It is a proxy solution to retrieve data from the WSE based on the Sys_Init, Query_Submit, and Query_Resp procedures. The Sys_Init procedure involves interest-based group creation. Query_submit corresponds to the process of sending queries to the WSE by setting a random path through multiple relay users. The Query_Resp occurs when the WSE receives the query, aggregates the profile of the user, and returns the answer to the user through TPs. However, the CoWSA does not explain how aggregated profiles are computed and what level of obfuscation a client achieves.

3. Profile Aware ObScure Logging (PaOSLo) Protocol

PaOSLo creates a single dynamic group of users with diverse interests to obscure a person's profile significantly. The PaOSLo consists of entities like users, a central server (CS), a query forwarding node (QFN), and a web search engine (WSE). A user is an individual who wishes to perform a web search secretly. A CS is a dedicated machine that oversees the PaOSLo group creation and execution process. QFN is one of the group users selected by the CS for sending queries of group users to WSE, retrieving and broadcasting the query results to group members.

3.1. PaOSLo Execution Process. The following are the steps necessary in the accomplishment of PaOSLo:

(i) The similarity between the users' profiles is computed in the first step of the PaOSLo execution. The user's profile construction is detailed in Section 3.

(ii) In the second step, the users are clustered using the K-mean algorithm based on their profile similarity.

(iii) In the following step, PaOSLo creates a group of users by selecting one from each cluster. This group will have users of different interests.

Figure 1 shows the activity diagram of computing the similarities between the users' profiles and grouping them into k clusters using the K-mean algorithm [8]. A user's profile is compared with the profiles of all other users using the cosine similarity metric [18]. An $n \times n$ matrix is obtained representing the similarities between the users' profiles. The matrix is uploaded into the Weka tool for clustering users using the K-mean algorithm.

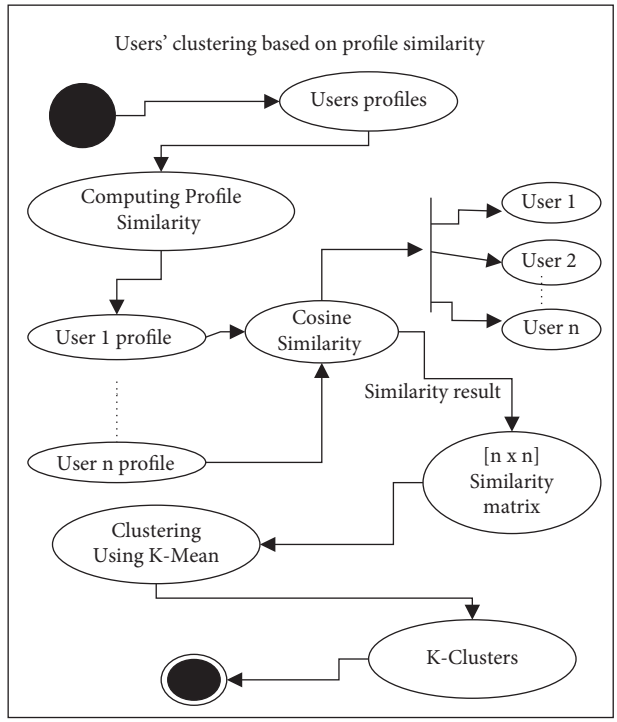

Figure 1: Activity diagram of computing similarity and users' clustering.

3.2. Group Making and Query Forwarding Node Selection Process. In this work, three group sizes have been chosen for experimentation, i.e., three users, four users, and five users. Figure 2 shows the activity diagram of group making and the QFN designation process. After clustering, one user from each cluster is selected by CS to create a group. The CS maintains a user_list[], containing the IP address and port number of the users selected from each cluster. Once the required group size is reached, the CS selects one user from the group as QFN. The CS forwards a "get_QFN_info" message to the user specified as QFN. When the user receives a get_QFN_info message, it generates a public-private key pair and selects a port number for communication. The QFN generates a detail_QFN message containing information about the encryption keys and port number for communication and forwards a detail_QFN message to the CS. The CS then broadcasts a user_list[] and detail_QFN in the created group. The role of QFN is to send queries of other group peers to the WSE, collect the queries' results from WSE, and broadcast them back to the group. The CS selects each user of the group as QFN in round-robin fashion. When all users play the role of QFN, the CS concludes the group. If there are "n" users in the group, the QFN forwards "$n-1$" queries to the WSE; the QFN never forwards their queries to the WSE.

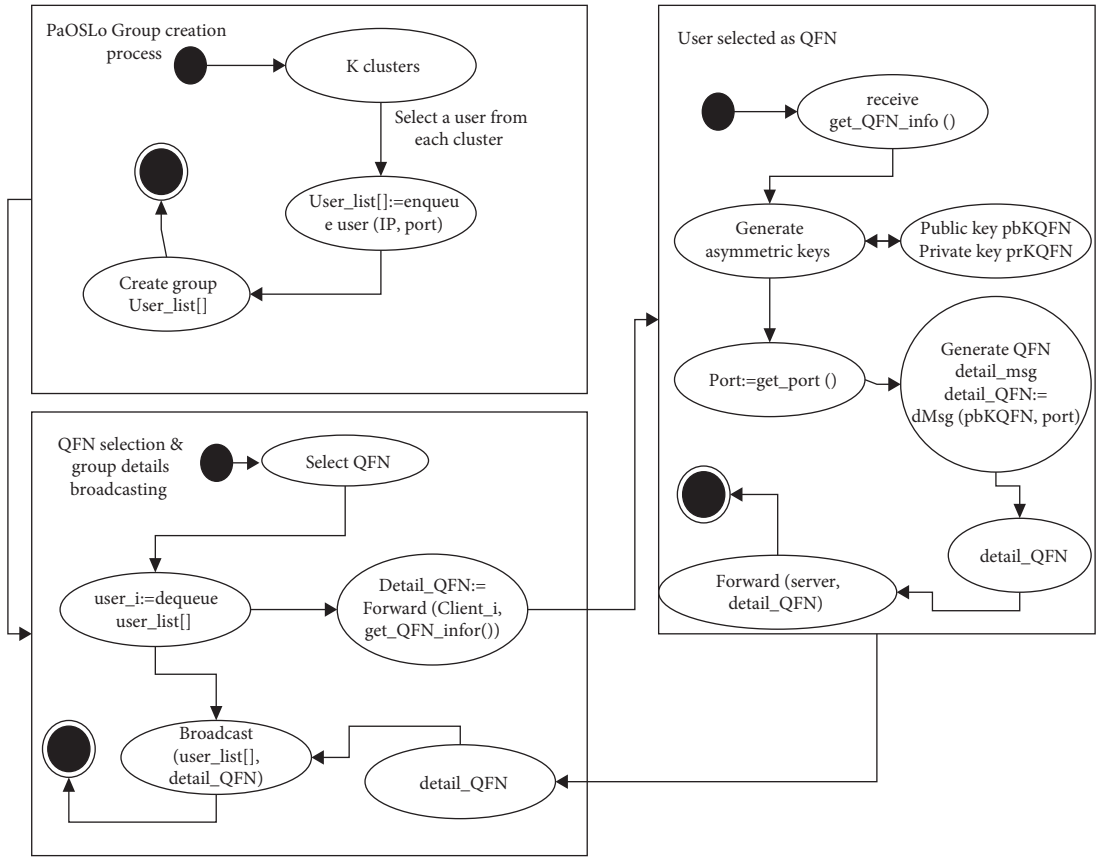

Figure 2: PaOSLo: activity diagram of group making and query forwarding node selection process.

3.3. Query Sending and Results Retrieval Procedure. Once the group making and QFN designation process concludes, the CS shares the user_list and detail_QFN message in the group. All users of the group receive this information. Figure 3 depicts the activity diagram of the query sending and result retrieval process. To send a query, a user (U_i) first gets information about the group users and the selected QFN. The user (U_i) creates a query and a cryptographic key (K_U_i). The Ui concatenates q and K_U_i making a QMsg. The U_i then encrypts QMsg with the public key of QFN, producing an encrypted query message (eQ). Afterwards, the U_i generates a q_ID, which the U_i will use for result identification. The U_i then generates the encrypted message (eMsg) by concatenating the encrypted message (eQ) and q_ID. The query encryption ensures the confidentiality of the query contents.

Once the query encryption process concludes, the eMsg is shuffled among the group users. The U_i flips a coin to decide where to forward the eMsg. If the coin produces a head, the U_i forwards the eMsg to QFN; if the coin produces a tail, the U_i forwards the eMsg to another random user U_j. The U_j on receiving eMsg does the same coin tossing and takes the same action on the results of head or tail. The shuffling of eMsg ensures that the query is unlinkable with the user. After a few passes, the eMsg reaches the QFN. The QFN in the first step separates the eMsg. The QFN gets all three parts distinctly (eQ, q_ID, and K_U_i). Afterwards, the QFN fetches the original query by decrypting the eQ with the private key. Afterwards, the QFN fetches the original query by decrypting the eQ with the private key. In the following step, QFN dispatches the query to WSE, which searches the relevant data over the Internet and delivers back the search query results (r). The QFN enciphers the results (r) with the cryptographic key (K_U_i) of the user. The QFN then appends q_ID with the encrypted results making an encrypted answer message (eAnsMsg). The QFN broadcasts the eAnsMsg to all group users. All users in the group receive the eAnsMsg. Each tries to match the q_ID to find out if the results are for their query. If the q_ID matches, the user decrypts the results (r), and the query sending process completes. Otherwise, the user drops the eAnsMsg.

4. Methodology

4.1. Dataset. America Online (AOL) released a three-month query log of 650 thousand users in 2006 for research [3, 19, 20]. It consisted of twenty million queries, but AOL did not inform the users about the release of the query log and that it would be freely obtainable [21]. To achieve the unlinkability between the users and queries, AOL had removed some information such as IP address, e-mail address, name, and other personal data from the query log. The query log is comprised of 5 features: "AnonID" describes the anonymous ID given to a user. Query, the actual search word of a user. Query Time, indicating the temporal information, ItemRank, and ClickURL, the URL clicked by the user after

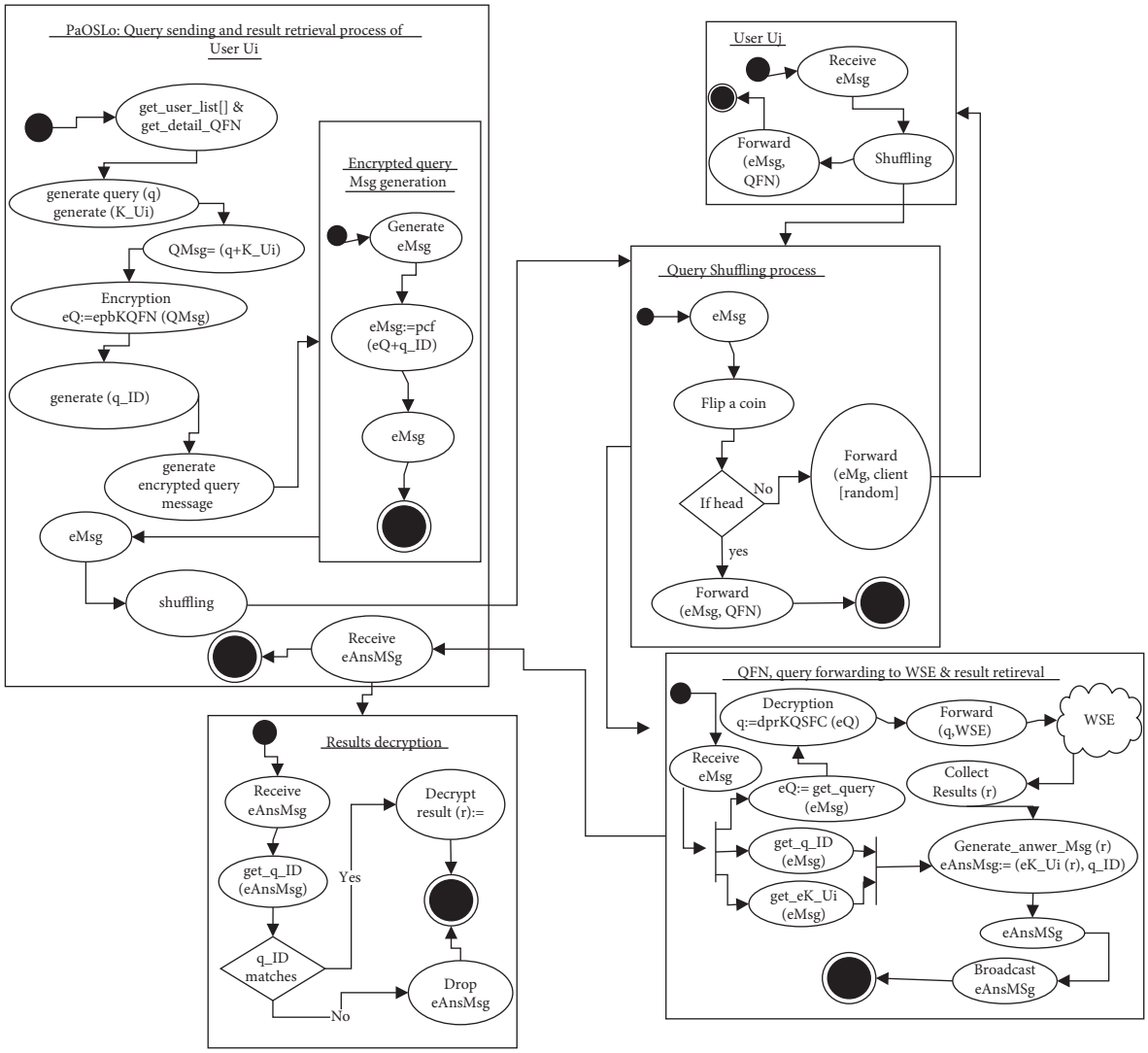

Figure 3: PaOSLo: activity diagram of sending and result retrieval process.

obtaining the results from AOL [19]. This query log is considered as a principal data source for evaluating web search privacy [2]. Piddinti and Saxena statistically examined various features of the AOL query log. The study shows that 98.72% of users have sent fewer than 100 queries over three months [2, 22]. 70% of people had fewer than thirty searches. In this work, we have selected a subset of the AOL query log dataset consisting of 1000 users ranging from highly active to minor active users for experimentation. By highly active, we mean those users who have sent more queries to the WSE, e.g., more than 200 queries. Similarly, minor active users are those who have sent less than 200 queries to the WSE. The selected users had sent a minimum of 25 queries to a maximum of 1514 queries. We have chosen the same dataset used by Ullah et al. [2] for the experiment. Table 1 shows the statistics of the users selected for the PaOSLo experimentation.

4.2. User Profile Construction. WSE builds the profile of the user from the queries it collects from the user. The WSE uses this profile to provide personalized results. Authors in [2, 3, 5, 11, 16, 19] have described the steps to build the user profile from the search queries. We have followed the same steps to create the user profile. Figure 4 shows the activity diagram of user profile construction. There are two significant steps taken in this process. This first step involves morphosyntactic analysis such as sentence detection, tokenization, part-of-speech tagging, and stop word removal for getting the query's actual topic defined by Cohen and Dolbey [23]. Semantic analysis is the second step of user profile construction. The words obtained by morphosynthetic are forwarded to Dmoz.org for user profile construction. Dmoz.org commonly referred to as the Open Directory Project (ODP) is an extensive Web directory managed by an alliance of volunteers [16, 24]. ODP classifies

Table 1: Dataset description.

Number of users in the dataset	1000
Total number of queries sent by the users	103644
Minimum-maximum queries by user	25–1512
Average queries by a user	103

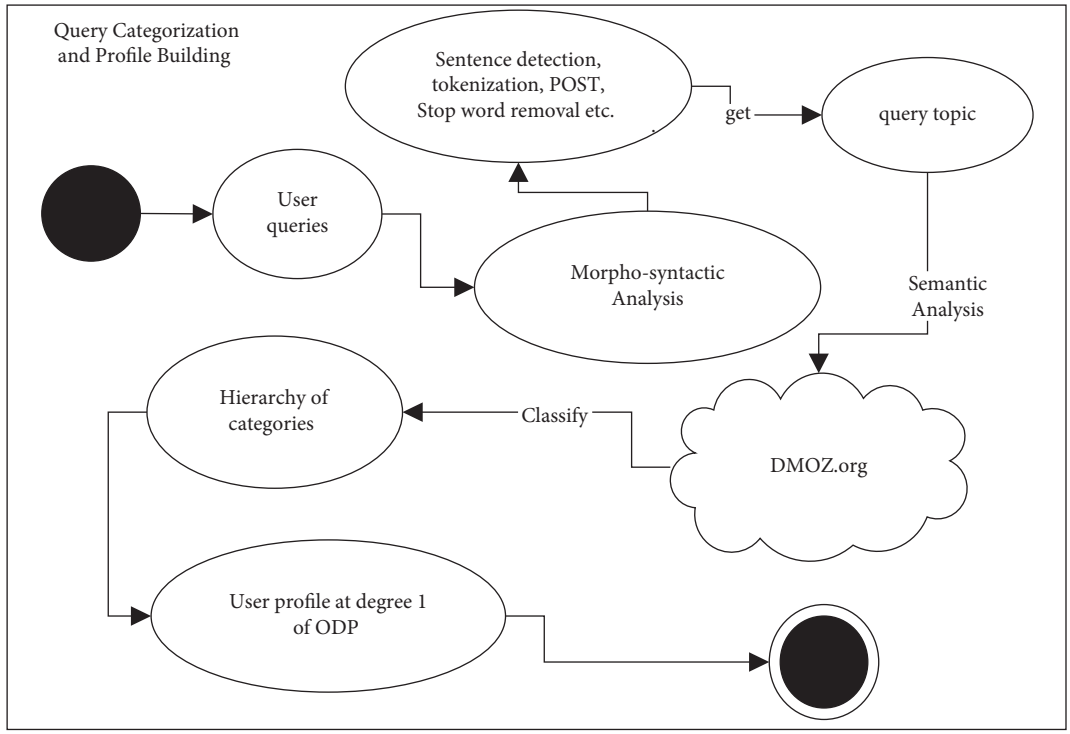

Figure 4: Activity diagram of user profile construction.

the search keywords into a hierarchy of levels. At the top level or first degree there are sixteen various classes. Each class has further subclasses, and so on and so forth. It has around 1 million distinctive classes [2, 16, 24]. Figure 5 shows the categories at first degree or top level of the ODP hierarchy. ODP categories any query into one of these 16 categories at degree 1. Table 2 shows a sample of query categorization by ODP. For example, a query "Valley National Banker" is categorized as "Business: Financial Services: Banking Services: Credit Unions: Regional: United States:" by the ODP. At degree 1, the query is represented as "business," at degree 2 as "Financial service," banking services at the third degree, and so on. By following the abovementioned steps, the profiles of all users are built.

4.3. Similarity Computation. After building the users' profiles, the categories/terms at degree 1 of a user profile are collected to compute the similarity between the users' profiles. Table 3 represents the term count at degree 1 of the ODP hierarchy of two sample users with AnonID "3978802" and AnonID "280617." The cosine similarity metric computes the likeness between two vectors. It is a function of the angle between the vectors of two users' profiles in the term vector space [26]. Equation (1) calculates the similarity between vectors A and B [6, 18]. The similarity function returns a value ranging from 0 to 1. The 0 represents distinct profiles, and the 1 donates the same matching profile. Based on the values of Tables 3 and 4, the similarity between user AnonID "3978802" and AnonID "280617" is 0.335. Following the same steps, the similarity between 1000 users is computed, and a 1000×1000 profile similarity matrix is made.

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{||A||||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{B_i^2}}. \quad (1)$$

4.4. Users Clustering. Once the similarity calculation process is completed, profile clustering, the second step of PaOSLo, is performed. The 1000×1000 profile similarity matrix is converted into Attribute-Relation File Format (ARFF) and loaded into the Weka tool for clustering. Using the K-mean algorithm, the Weka tool clusters the users based on profile similarity into three clusters, four clusters, and five clusters [8]. The distance between the objects (users' profiles) is determined by the Euclidean distance. If the values of Euclidean distance between the users' profiles are low, the users are assigned to the same cluster, but they are different in other cases.

Table 4 shows the count of users allotted to each cluster. It is calculated over the term count of degree 1 categories of the ODP hierarchy.

Figure 6 shows a term count in each category of a first degree in ODP hierarchy. The X-axis of the graph's x-axis shows the top level of 16 categories of ODP, whereas the y-axis shows the count of the terms in each cluster. Cluster 1 contains those users who have frequently sent queries from the "Regional" category. Cluster 2 has users who have sent maximum queries from the "Business" category. Furthermore, cluster 3 has users who mostly sent queries from the "Arts" category. Similarly, Figure 7 shows that when users are grouped into four clusters, cluster 1 contains users who have forwarded queries from the "Arts" category. Cluster 2 contains users interested in "Business" queries; cluster 3 contains

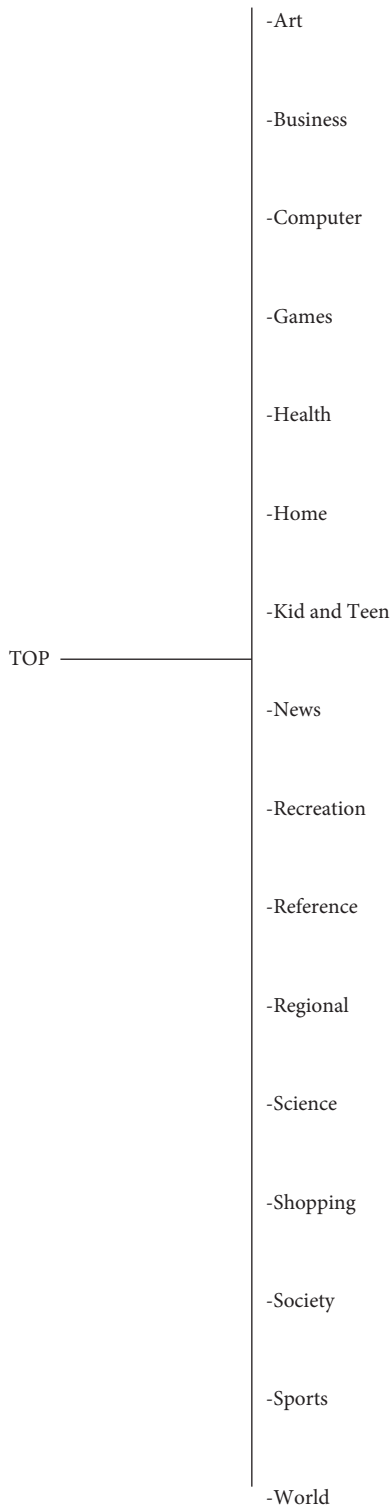

Figure 5: Top level categories of the ODP hierarchy [25].

users of "Regional" interests; and cluster 4 of the "Computer" category.

Figure 8 shows when users are grouped into five clusters based on the term count of degree 1 of the ODP hierarchy. Cluster 1 contains users who have sent maximum queries from the "Arts" category. "Computer" related queries dominate cluster 2, cluster 3 with "Regional" interests, and cluster 4 with "Recreation," while cluster 5 users are interested in "Business" queries.

5. PaOSLo Privacy Evaluation

The primary goal of a user simulating a distributed privacy-preserving protocol is to accomplish the following objectives:

(i) A user aims to attain confidentiality, i.e., the query content and result to query remains hidden from the group entities

(ii) A user intends to achieve unlinkability such that a query cannot be linked back to the user

(iii) A user endeavors to accomplish indistinguishability such that the WSE shall not create the actual profile

Local privacy and profile privacy are the aspects that researchers have described to assess the privacy of distributed protocols [2, 11, 16, 20]. Local privacy computes the probability of associating a query with the user by a curious entity of PaOSLo. Furthermore, profile privacy estimates the extent of user profile obfuscation.

A user executing PaOSLo achieves local privacy through confidentiality, query shuffling, and profile privacy by forwarding other users' queries. A user attains the confidentiality of a query and results as both the query and results are encrypted. The query encryption is performed using the RSA encryption algorithm. As mentioned earlier, the user encrypts the query (q) with the public key of QFN. No group user will be able to see the query content. Similarly, the QFN encrypts the results with the user's cryptographic key (K_Ui). We have used the AES encryption method for result (r) encryption. In such a case, the query and results will remain concealed from users of the group peers. Furthermore, to break the connection concerning a user and a query, it is shuffled among the peers. Even when the QFN receives a query from the user, it will not be able to determine if the query they received from the user is the query's originator or just a forwarder.

Additionally, as a user forwards the queries of group peers, their profile retained by WSE will be obfuscated with the diverse interests of the group peers. In such a case, the user achieves indistinguishability. The following section presents a thorough assessment of local privacy and profile privacy.

5.1. Local Privacy. To compute the local privacy, we must calculate the probability of associating a query by a curious entity with the originator. Let two random variables, Sr and Py, where Sr denotes the source of the query, and Py represents the proxy (a user in the group) that passes the query to the QFN. Suppose there are "n" users in the group. If the QFN wants to find the originating user of a suspicious query (q), the chances of associating a (q) to "Ui" are stated as follows:

Table 2: Example of query classification by ODP.

Query	ODP classification at different degrees
Valley National Banker	Business: financial services: banking services: credit unions: regional: United States
Photography Studios	Arts: photography: techniques and styles: documentary: photographers
mac.com	Computers: software: operating systems: MacOS: Internet

Table 3: Terms extracted for degree 1 of users AnonID "3978802" and AnonID "280617."

Query class	User 3978802	User 280617
Arts	2	0
Business	2	6
Computers	2	5
Games	0	1
Health	7	0
Home	0	3
Kids and teen	1	0
News	0	2
Recreation	1	0
Reference	1	1
Regional	1	4
Science	0	0
Shopping	0	0
Society	1	1
Sports	0	0
World	3	0

Table 4: Number of users in each cluster after K-mean clustering.

Cluster counts	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Three clusters	390	306	304	—	—
Four clusters	266	188	311	235	—
Five clusters	241	122	222	236	179

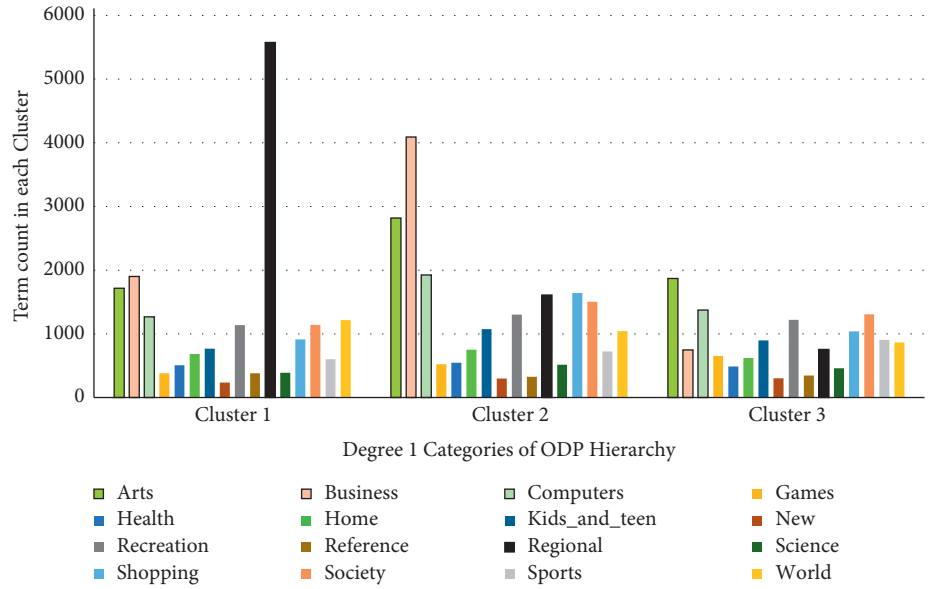

Figure 6: Term count in three clusters.

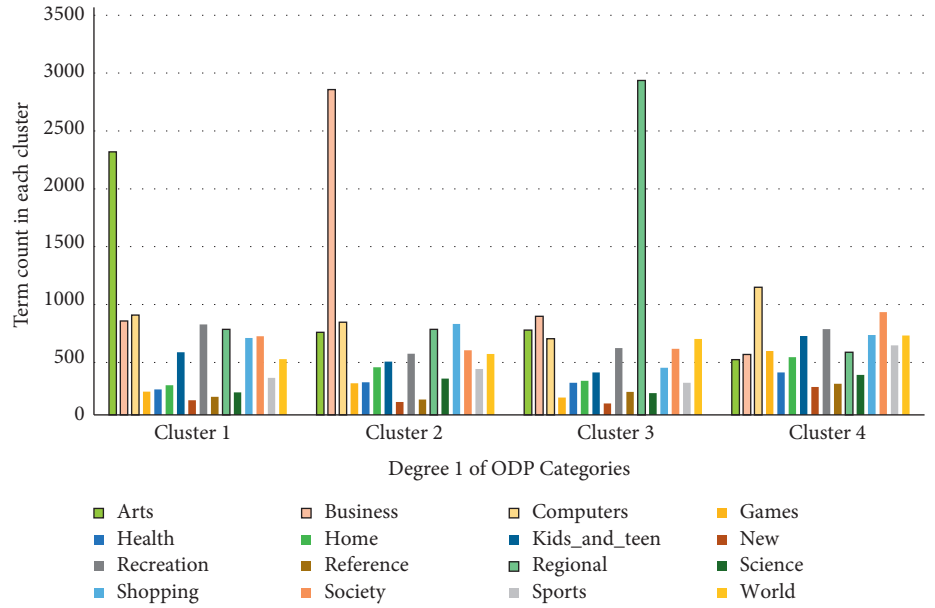

Figure 7: Term count in four clusters.

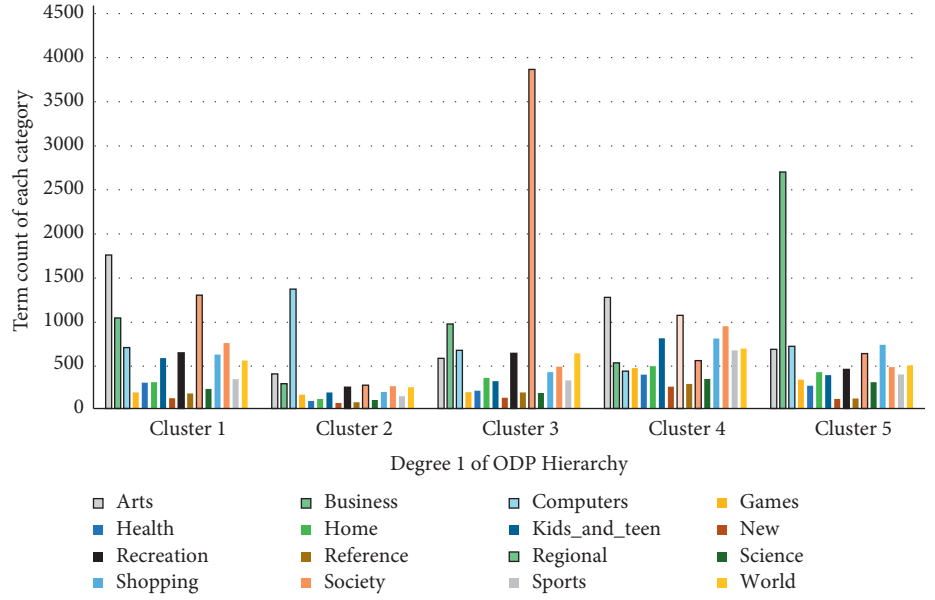

Figure 8: Term count in five clusters.

$$\Pr[\mathrm{Sr} = \mathrm{Ui} \mid \mathrm{Py} = \mathrm{Uj}] = \frac{\Pr[\mathrm{Py} = \mathrm{Uj} \mid \mathrm{Sr} = \mathrm{Ui}] \cdot \Pr[\mathrm{Sr} = \mathrm{Ui}]}{\Pr[\mathrm{Py} = \mathrm{Uj}]}. \tag{2}$$

Equation (2) represents the probability of source when the proxy is given to curious entity.

$$\Pr[\mathrm{Py} = \mathrm{Uj} \mid \mathrm{Sr} = \mathrm{Ui}] = \Pr[\mathrm{Py} = \mathrm{Uj}]. \tag{3}$$

$$\Pr(\mathrm{Sr} = \mathrm{Ui}) = \frac{1}{n-1}. \tag{4}$$

Equation (4) represents the probability of source Ui where "n" represents the number of users in a group and i, $j \in (1, \ldots, n)$, as QFN is not the query source, QFN excludes itself.

$$\Pr(\mathrm{Sr} = \mathrm{Ui} \mid \mathrm{Py} = \mathrm{Uj}) = \frac{1}{n-1}. \tag{5}$$

Equation (5) shows that the probability of associating a query (q) with Ui by QFN depending on the count of users in a group. However, if QFN and C users collaborate to identify the query (q) originator, then the chance of associating (q) is given in.

$$\Pr(\mathrm{Sr} = \mathrm{Ui} \mid \mathrm{Py} = \mathrm{Uj}) = \frac{1}{n-C}. \tag{6}$$

Table 5: Average PEL comparison of UUP(e), OSLo, and PaOSLo.

Number of users	Protocol	Degree1	Degree 2	Degree 3	Degree 4
3 users	UUP(e)	51.86	13.39	7.3	7.22
	OSLo	47.7	12.79	6.95	7.17
	PaOSLo	46.65	11.14	5.5	6.01
4 users	UUP(e)	51.16	13.14	7.07	7.2
	OSLo	48.56	12.76	6.85	6.95
	PaOSLo	46.32	10.6	5.44	5.75
5 users	UUP(e)	51.55	13.47	7.37	7.26
	OSLo	49.18	12.86	6.99	6.85
	PaOSLo	46.29	10.58	5.39	5.92

Equation (6) shows that if QFN makes a coalition with C users, the chance of associating (q) with an initiator $1/(n - C)$ which means all compromised C users will be excluded from the list. The value of n-C must be greater than 1.

5.2. Profile Privacy. Profile privacy measures the user's privacy in comparison to the WSE. Equation (7) shows a profile exposure level (PEL). It is an estimation metric utilized to evaluate the profile privacy. PEL computes the variation between an original profile and an obscured profile of an individual. The difference between the user's profiles has been calculated using entropy and mutual information [2, 11, 16]. It is necessary to mention that the original profile is made from the queries sent directly to WSE. In contrast, the obfuscated profile is built from the queries after the execution of PaOSLo.

$$\text{PEL} = \left(\frac{I(P,Q)}{H(P)} \times 100 \right). \tag{7}$$

$H(P)$ represents the entropy, whereas $I(P, Q)$ denotes the mutual information. The value of PEL is between 0 and 100, where 100 means fully exposed and 0 means no exposure.

6. Results and Discussion

This segment provides a comprehensive explanation of simulations conducted for the estimation of the profile privacy of users by executing PaOSLo. To perform experiments, we have developed a Java-based program to simulate PaOSLo using java socket programming. In the first step, the users get connected to the CS, and then the CS creates groups by following the steps mentioned in Section 3. We have adopted a CryptoUtil key pair generator method to generate RSA public-private key pairs and AES encryption key. The simulation of PaOSLo is performed over Intel i5-11400F CPU with 8 Gb RAM over Windows 10. After the simulation process, each user's query log is obtained and developed their profile from it. The profile privacy a user succeeds by executing PaOSLo has been compared with modern distributed privacy-preserving protocols such as OSLo [11] and UUP(e) [16].

Table 5 shows the average PEL a user achieves by executing PaOSLo. The average PEL at ODP's first degree for the group consisting of three users is 46.56%. Likewise, at degree 2, the average PEL drops to 11.14%, the average PEL further drops to 5.5% at degree 3, and so on. Similarly, for a group count of four, the average PEL is 46.32% at degree 1, 10.6% at degree 2, 5.44% at degree 3, and 5.75% at the fourth degree ODP. Likewise, the results at degree 1 for the group count of five is 46.29%, 10.58%, 5.83%, and 5.92% at the second, third, and fourth degrees of the ODP hierarchy, respectively. On the other hand, the results show that the OSLo provided 47.7% average PEL, and UUP (e) depicted 51.86% average PEL for the group size of three users at degree 1 of the ODP hierarchy. The average PEL values of UUP (e) dropped 51.16%, whereas the OSLo value reached 48.56% for the group size of four users. Similarly, when the group size is increased to five users, the UUP (e) showed 51.55% and OSLo showed 49.19% average PEL at degree 1 of the ODP hierarchy. The results show that the average PEL of a user simulating PaOSLo decreases when the group size is increased. A similar pattern is observed at all degrees of the ODP hierarchy. Such a decrease is because the user's profile is obfuscated by multiple users having different interests. However, in a random grouping, the chances of getting grouped with users having similar interests are higher, resulting in lower-profile obfuscation or higher profile exposure.

In this study, the profile privacy a user achieves after executing PaOSLo is compared with the modern privacy-preserving protocol UUP(e) and OSLo. This comparison is between a random grouping of users and a profile-aware grouping. Figure 9 shows that PaOSLo has less average PEL at all degrees of the ODP hierarchy than UUP(e) and OSLo for any group size. The PaOSLo has 10% less average PEL than UUP(e) and 2.5% less than OSLo at first degree of ODP for the group count of three members. The PaOSLo has 9.6% and 4.7% smaller average PEL than UUP(e) and OSLo at first degree of the ODP for the group of four members. Similarly, for five users group, the PaOSLo has 8.7% and 4.36% improved profile privacy compared to the abovementioned protocols. A similar effect is observed at a higher degree of ODP hierarchy, the PaOSLo outperform their counterpart OSLo and UUP(e). Results show that PaOSLo preserved better profile privacy than UUP(e) and OSLo at all degrees of ODP for any group size. The reason for the PaOSLo better results is the fact that it first clusters users based on their profile similarities compared to the random grouping. In such a case, the group created by CS from clusters contains users of different interests. Each user forwards a query of users from the other clusters; hence, a user's profile is

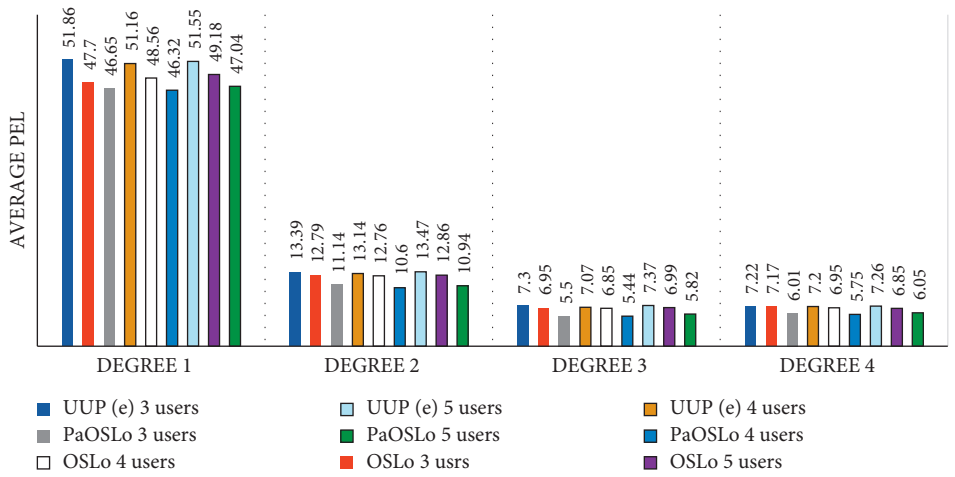

Figure 9: Average PEL of PaOSLo vs. OSLo vs. UUP(e).

Table 6: List of abbreviations and acronyms used in this article.

Abbreviation	Explanations	Abbreviation	Explanations
WSE	Web search engine	QMsg	Query message
PaOSLo	Profile Aware ObScure Logging	eQ	Encrypted query message
ODP	Online directory project	eMsg	Encrypted message
TOR	The onion routing	r	Results
CS	Core Server	eAnsMsg	Encrypted answer message
UPIR	User private information retrieval	AOL	America Online
TP	Third party	AnonID	Anonymous ID
QFN	Query forwarding node	Pr	Probability
Q	Query	S	Source
Ui	User i	P	Proxy
Uj	User j	PEL	Profile exposure level
K_Ui	Encryption key of Ui	QMsg	Query message

obfuscated with diverse interests' multiple peoples, and thus, the user achieves maximum obfuscation. Table 6 shows the list of abbreviations and acronyms used in this article.

7. Conclusion and Future Work

The existing distributed privacy-preserving protocols create a group of "*n*" users on a first-come, first-serve basis. Users are randomly grouped to forward each other's queries to the WSE without any prior knowledge of their interests. In random grouping, there is a greater chance that users having similar interests may be group together. Such random grouping has a trivial effect on profile obfuscation. To overcome the limitation of random grouping on web search privacy, this study recommends a novel distributed privacy-preservation protocol, PaOSLo. It obfuscates a person's profile with queries of those who have different interests. Systematic grouping is the primary objective of this work. Profile building and finding the similarity between users' profiles are the fundamental steps of PaOSLo. The measures the resemblance between the users' profiles cosine similarity is used. The K-mean algorithm clusters the users into three, four, and five clusters based on the similarity computed in the previous step. We experimented with and evaluated the impact of systematic grouping and compared random grouping with three members, four members, and five members. Results show that the systematic group depicts better profile privacy compared to the random group.

In the future, we are interested in investigating the impact of systematic grouping on the quality of the results. The impact of profile-aware grouping on the search results and the delay caused by clustering needs to be measured in the future. The profile privacy shall also be examined with other privacy metrics such as entropy, standard deviation, and KL divergence. Furthermore, maintaining privacy and personalization at the same time will be the future direction of this work. [27].

Abbreviations

WSE: Web search engine
PaOSLo: Profile Aware ObScure Logging
ODP: Online directory project
TOR: The onion routing
CS: Core Server
UPIR: User private information retrieval
TP: Third party
QFN: Query forwarding node
q: Query
U_i: User i

U_j: User j
K_Ui: Encryption key of Ui
QMsg: Query message
eQ: Encrypted query message
eMsg: Encrypted message
r: Results
eAnsMsg: Encrypted answer message
AOL: America Online
AnonID: Anonymous ID
Pr: Probability
S: Source
P: Proxy
PEL: Profile exposure level
QMsg: Query message.

Data Availability

The datasets used for the experimentation in this work are accessible at https://github.com/mrmohibkhan/Profile-aware-OSLo.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank SAUDI ARAMCO Cybersecurity Chair at Imam Abdulrahman Bin Faisal University (IAU) for supporting and funding this work.

References

[1] N. Kaaniche, S. Masmoudi, S. Znina, M. Laurent, and L. Demir, "Privacy preserving cooperative computation for personalized web search applications," in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pp. 250–258, ACM, Brno Czech Republic, March 2020.

[2] M. Ullah, R. Khan, M. Inam Ul Haq et al., "Multi-group ObScure logging (MG-OSLo) A privacy-preserving protocol for private web search," *IEEE Access*, vol. 9, pp. 79005–79020, 2021.

[3] M. Rodriguez-Garcia, M. Batet, D. Sánchez, and A. Viejo, "Privacy protection of user profiles in online search via semantic randomization," *Knowledge and Information Systems*, vol. 63, pp. 1–23, 2021.

[4] A. Kumar and K. Hosanagar, "Measuring the value of recommendation links on product demand," *Information Systems Research*, vol. 30, no. 3, pp. 819–838, 2019.

[5] R. Khan, M. Ullah, A. Khan, M. I. Uddin, and M. Al-Yahya, "NN-QuPiD attack: neural network-based privacy quantification model for private information retrieval protocols," *Complexity*, vol. 2021, Article ID 6651662, 8 pages, 2021.

[6] R. Khan, M. A. Islam, M. Ullah, M. Aleem, and M. A. Iqbal, "Privacy exposure measure: a privacy-preserving technique for health-related web search," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 6, pp. 1196–1204, 2019.

[7] A. Raza, K. Han, and S. O. Hwang, "A framework for privacy preserving, distributed search engine using topology of DLT and onion routing," *IEEE Access*, vol. 8, pp. 43001–43012, 2020.

[8] J. Wu, *Advances in K-Means Clustering: A Data Mining Thinking*, Springer Science & Business Media, Berlin, Germany, 2012.

[9] J. Parra-Arnau, D. Rebollo-Monedero, and J. Forné, "Measuring the privacy of user profiles in personalized information systems," *Future Generation Computer Systems*, vol. 33, pp. 53–63, 2014.

[10] M. K. Reiter and A. D. Rubin, "Crowds: anonymity for web transactions," *ACM Transactions on Information and System Security*, vol. 1, no. 1, pp. 66–92, 1998.

[11] M. Ullah, M. A. Islam, R. Khan, M. Aleem, and M. A. Iqbal, "ObSecure Logging (OSLo): a framework to protect and evaluate the web search privacy in health care domain," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 6, pp. 1181–1190, 2019.

[12] J. Domingo-Ferrer, M. Bras-Amo'ros, Q. Wu, and J. Man'jon, "Userprivate information retrieval based on a peer-to-peer community," *Data & Knowledge Engineering*, vol. 68, no. 11, pp. 1237–1252, 2009.

[13] C. M. Swanson and D. R. Stinson, "Extended combinatorial constructions for peer-to-peer user-private information retrieval," *Advances in Mathematics of Communications*, vol. 6, 2011.

[14] C. M. Swanson and D. R. Stinson, "Extended results on privacy against coalitions of users in userprivate information retrieval protocols," *Cryptography and Communications*, vol. 7, no. 4, pp. 415–437, 2015.

[15] J. Castellá-Roca, A. Viejo, and J. Herrera-Joancomartì, "Preserving users privacy in web search engines," *Computer Communications*, vol. 32, no. 13-14, pp. 1541–1551, 2009.

[16] C. Romero-Tris, J. Castella-Roca, and A. Viejo, "Distributed system for private web search with untrusted partners," *Computer Networks*, vol. 67, pp. 26–42, 2014.

[17] J. Domingo-Ferrer, S. Martínez, D. SÆnchez, and J. Soria-Comas, "Coutility: self-enforcing protocols for the mutual benefit of participants," *Engineering Applications of Artificial Intelligence*, vol. 59, pp. 148–158, 2017.

[18] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008)*, vol. 4, pp. 9–56, Christchurch, New Zealand, 2008.

[19] R. Khan, A. Ahmad, A. O. Alsayed, M. Binsawad, M. A. Islam, and M. Ullah, "Qupid Attack: machine learning-based privacy quantification mechanism for PIR protocols in health-related web search," *Scientific Programming*, vol. 2020, Article ID 8868686, 11 pages, 2020.

[20] R. Masood, D. Vatsalan, M. Ikram, and M. A. Kaafar, "Incognito: a method for obfuscating web data," in *Proceedings of the 2018 World Wide Web Conference*, pp. 267–276, ACM, Lyon, France, April 2018.

[21] D. Pàmies-Estrems, J. Castellà-Roca, and J. Garcia-Alfaro, "A real-time query log protection method for web search engines," *IEEE Access*, vol. 8, pp. 87393–87413, 2020.

[22] S. T. Peddinti and N. Saxena, "On the privacy of Web search basedon query obfuscation: a case study of trackmenot," in *Proc. Int. Symp. Privacy Enhancing Technol. Symp.*, pp. 19–37, Springer, Berlin, Germany, 2010.

[23] K. B. Cohen and A. Dolbey, "Foundations of statistical natural language processing," *Language*, vol. 78, no. 3, p. 599, 2002.

[24] N. C. Senthilkumar and Ch Pradeep Reddy, "Prediction of user interest fluctuation using fuzzy neural networks in web search," *International Journal of Intelligent Unmanned Systems*, vol. 8, no. 4, pp. 307–319, 2020.

[25] A. Viejo, J. Castella-Roca, O. Bernadó, and J. M. Mateo-Sanz, "Single-party private web search," in *Proceedings of the 2012 Tenth Annual International Conference on Privacy, Security and Trust*, pp. 1–8, IEEE, Paris, France, July 2012.

[26] A. Huang, "Similarity measures for text document clustering," in *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC 2008)*, vol. 4, pp. 9–56, Universities and Research Institutions, Christchurch, New Zealand, April 2008.

[27] M. Juarez and V. Torra, "Dispa: an intelligent agent for private web search," in *Advanced Research in Data Privacy*, pp. 389–405, Springer, New York, NY, USA, 2015.