JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Query Obfuscation Scheme for Private Web Search Using Proxy-Query Based Approach

A. Sudhasri*[1], R. Surekha[2], V. Sai Shivani[3] , P.Keerthana[4], D.Saidulu [5]

[*1] UG Student, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana,India
[2] UG Student, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana,India
[3] UG Student, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana,India.
[4] UG Student, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana,India.
[5]Associate Professor, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana,India.

Abstract : Web search engines are essential tools for accessing information on the internet, but concerns about privacy arise due to the storage of sensitive user data in query logs. To address this, various private web search (PWS) schemes have emerged, aiming to safeguard user privacy while still providing effective search results. However, there's a lack of clarity on which characteristics are crucial for ensuring web search privacy (WSP) in PWS schemes. It introduces a new PWS scheme employing proxy-query-based query obfuscation, a novel approach in PWS research. By utilizing proxy queries, users can retrieve information from search engines without directly exposing their true queries, enhancing privacy. The second part of this paper focuses on delineating key characteristics of PWS and evaluating modern PWS schemes against these criteria. Through analysis, it becomes evident that only the newly proposed PWS scheme fulfills all identified characteristics. Existing PWS systems are found to be vulnerable to WSP attacks due to their failure to meet all essential PWS characteristics.

***IndexTerms* - Proxy-query,Web search,Information System,Web search privacy,Private web search**

I. Introduction

Personalized web search, a pivotal feature of contemporary search engines, tailors search results to individual users based on various contexts. Achieved through personalized information retrieval (PIR) techniques, this customization extracts personal data from query logs to infer user preferences. While enhancing retrieval efficacy, PIR raises significant privacy concerns, as evidenced by a Pew survey where approximately 75% of users express dissatisfaction with search engines tracking their queries for personalization.[1]. Current privacy protection techniques predominantly focus on identifiability aspects, emphasizing secure communication and encrypted data storage. However, the critical factor of linkability, often overlooked, plays a crucial role in web search privacy. Through linkability, search engines can deduce detailed user interests by connecting multiple queries, diminishing user control over their privacy. Research has demonstrated the accuracy of classifiers in linking related queries to users, highlighting the vulnerability of current privacy measures.

To illustrate the ramifications of inadequate web search privacy, consider a scenario where a user seeks information on sensitive topics like depression, HIV, or pregnancy. Exposure of such queries through data sale or log compromise can lead to exploitation of personal health information by malicious entities, reminiscent of the 2006 AOL query log breach reported by The New York Times. The breach unveiled private health concerns of numerous users, exemplified by a 62-year-old lady's search for information on hand tremors, dry mouth, and nicotine effects.[2]

In response, we propose a private web search (PWS) scheme utilizing proxy-query-based query obfuscation to safeguard web search privacy. This emerging approach provides an information retrieval (IR) framework using proxy queries, wherein users issue proxy queries rather than true ones. The IR system generates cover queries from proxy queries, obscuring user intent. A key challenge lies in ensuring the plausibility of generated cover queries, which our scheme addresses by maintaining plausibility not only in the current query but also in the sequence of previous queries.

II. EXISTING SYSTEMS

The previous scheme (ProxyTermPWS) on this research achieves the above objective by mapping the terms of topics containing sensitive information with the terms of proxy topics. The key weakness of ProxyTermPWS is that it develops proxy-term mapping between individual terms of proxy and cover topics. This provides low effectiveness if the queries contain more than one term. This is due to the computational difficulty of achieving optimum proxy-term mapping for every combination of valid query terms.

EXISTING SYSTEM DISADVANTAGES

Furthermore, the existing approach has a drawback in that it provides less effectiveness for query obfuscation when a user issues a series of consecutive inquiries linked to a similar topic.

I. RESULTS AND DISCUSSION

This project is implements like web application using COREJAVA and the Server process is maintained using the SOCKET & SERVERSOCKET and the Design part is played by Cascading Style Sheet.The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations[3] and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

II. DEVELOPING METHODOLOGIES

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies[4].

PROPOSED SYSTEM

If utilized against an individual, the query log may pose privacy concerns and reveal a lot of information about them. In recent years many private web search (PWS) schemes have been proposed to realize privacy-preserving web search. Although each PWS scheme claims to have unique features for attaining web search privacy (WSP), no study explains which private web search characteristics should be considered when building and utilizing a PWS scheme. There are two objectives of this article. [5]In the first part of the article, we present a novel PWS scheme that uses a proxy-query-based query obfuscation approach. The challenges we want to address in the article include identifying the computing cost of searching for an ideal proxy-query mapping and proposing a heuristic for searching for an optimal mapping. Once the proposed scheme has discovered the best mapping, it makes the mapping available to all users in the form of a proxy-query dictionary[6].

PROPOSED SYSTEM ADVANTAGES

- More Security.
- It more efficiency and authentication service.
- Furthermore, as each proxy group contains a valid set of proxy and cover queries, the proposed technique does not generate an exhaustive set of cover queries for processing and recovering true query.
- an ideal proxy-query mapping and proposing a heuristic for searching for an optimal mapping

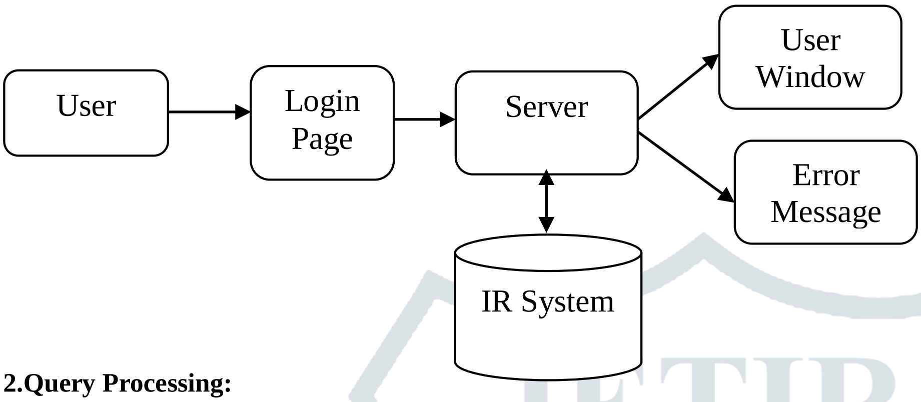
III. METHODOLOGIES

[7]To retrieve information from the mapped topics privately, users transform their true queries with proxy queries and issue to the IR system. When the proxy query is received, the IR system retrieves the results of all cover and proxy-query and delivers them to user. The IR system cannot identify real query in this manner. The user's machine only displays the result of true query's and ignores all other queries. The suggested approach is well suited to personalized IR settings, which have received much research and

are frequently employed in commercial search engines In personalized IR, search engines store users' queries in query logs to improve future queries' search effectiveness.

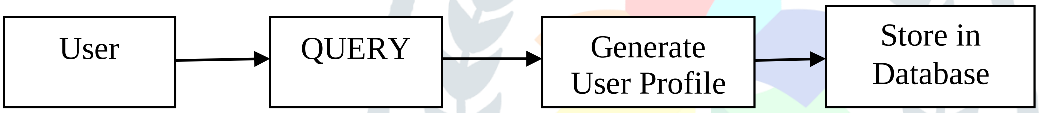
1.User Interface Design:

To connect with server user must give their username and password then only they can able to connect the server. If the user already exits directly can login into the server else user must register their details such as username, password and Email id, into the server. Server will create the account for the entire user to maintain upload and download rate.



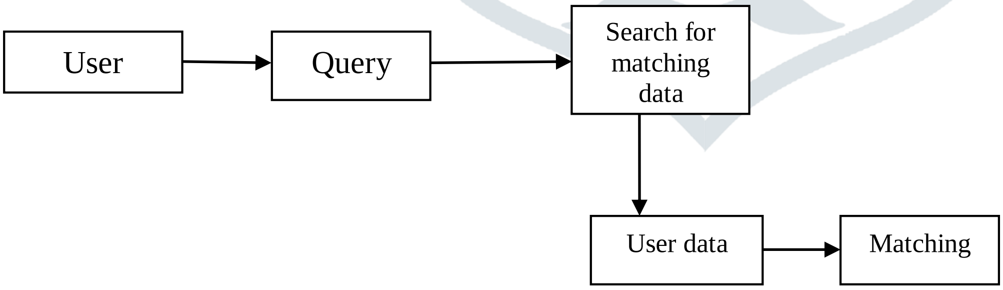
2.Query Processing:

In this module, the data is given by user requests goes to server, When a user issues a query on the client, the proxy generates a user profile in runtime in the light of query terms. The output of this step is a generalized user profile satisfying the privacy requirements. The generalization process is guided by considering two conflicting metrics, namely the personalization utility and the privacy risk, both defined for user profiles were administrator maintains all files and responsible for storing that files into cloud.



3.Combining User Profile and Proxy Query (Similarity Computation):

In this model, user given query and the generalized user profile are sent together to the server for search. Query with related user preferences stored in a user profile with the aim of providing better search results.

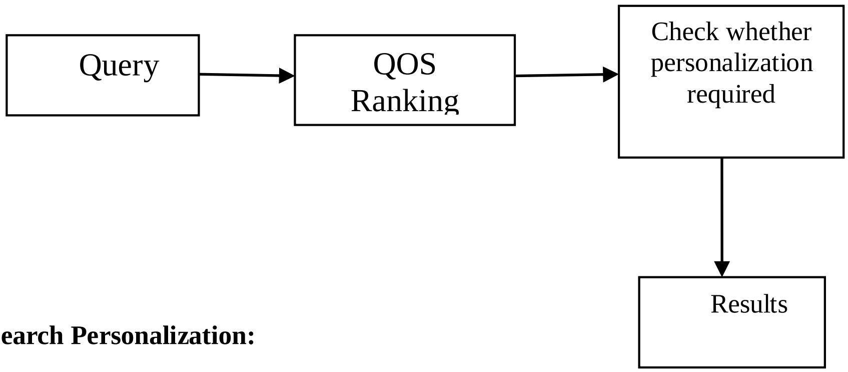


4.Online Generalization or IR System

The IR system cannot identify real query in this manner. The user's machine only displays the result of true query's and ignores all other queries. The suggested approach is well suited to personalized IR settings, which have received much research and are frequently employed in commercial search engines. In personalized IR, search engines store users' queries in query logs to improve future queries' search effectiveness.

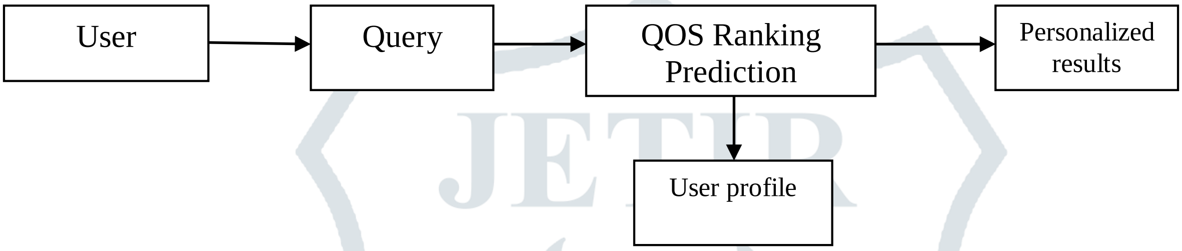
5.QOS Ranking Prediction:

In this model, user given query based on privacy requirements and cost of profiling search results are checked with the help of QOS & PREDICTION ACCURACY protocols whether to personalize or not.



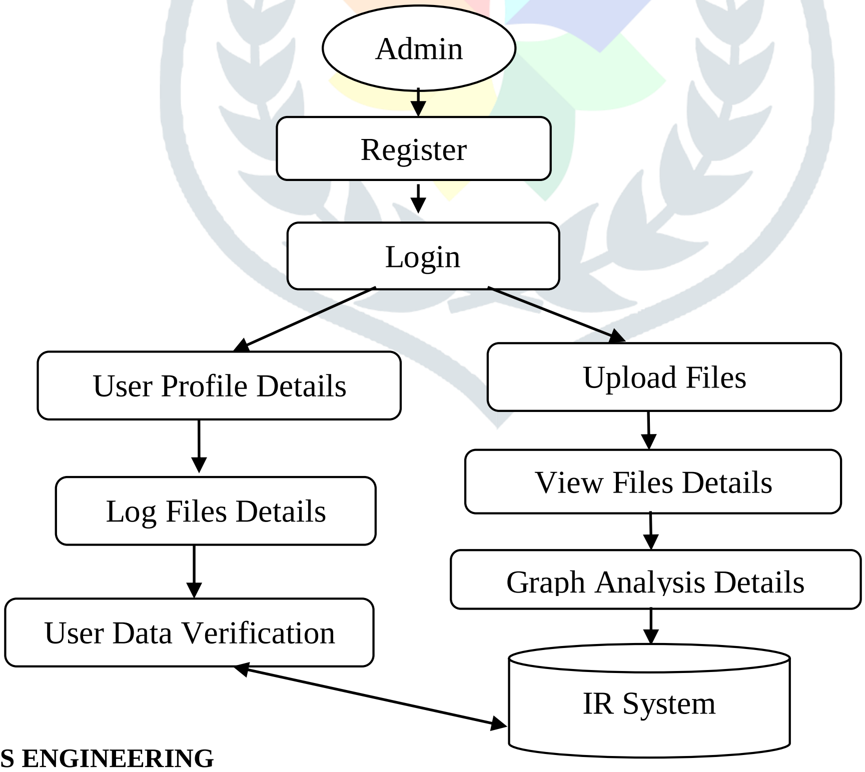
6.**Search Personalization:**

In this model user given query search results are personalized according to user profile and delivered back to the query proxy. After results are shown to user.



7.**Admin**

In this model Admin login with help of name & password. After login he/she having some option like user profiles, upload files with help of all details, view uploaded files data, search log files details & analysis of users searching query details. Admin maintain all details of user & some other process also.



REQUIREMENTS ENGINEERING

These are the requirements for doing the project. Without using these tools & software's we can't do the project. So we have two requirements to do the project. They are

- Hardware Requirements.
- Software Requirements

1.HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system.[8] They are used by software engineers as the starting point for the system design. It shoulds what the system and not how it should be implemented.

• PROCESSOR	:	PENTIUM IV 2.6 GHz, Intel Core 2 Duo.
• RAM	:	512 MB DD RAM
• MONITOR	:	15" COLOR
• HARD DISK	:	40 GB

2. SOFTWARE REQUIREMENTS

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity.

• Front End	:	J2EE (JSP, SERVLET)
• Back End	:	MY SQL 5.5
• Operating System	:	Windows 7
• IDE	:	Eclipse

3. FUNCTIONAL REQUIREMENTS

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behaviour, and outputs.[9] The outsourced computation is data is more secured.

- User interface.
- Query processing.
- Combining User profile and query
- Online Generalization
- Search personalization
- Admin

4. NON-FUNCTIONAL REQUIREMENTS

The major non-functional Requirements of the system are as follows.

- **Usability** The system is designed with completely automated process hence there is no or less user intervention.
- **Reliability** The system is more reliable because of the qualities that are inherited from the chosen platform java. The code built by using java is more reliable.
- **Performance** This system is developing in the high-level languages and using the advanced front-end and back-end technologies it will give response to the end user on client system with in very less time.
- **Supportability** The system is designed to be the cross platform supportable. The system is supported on a wide range of hardware and any software platform, which is having JVM, built into the system.
- **Implementation** The system is implemented in web environment using struts framework. The apache tomcat is used as the web server and windows xp professional is used as the platform. Interface the user interface is based on Struts provides HTML Tag.

IV. SYSTEM ARCHITECTURE

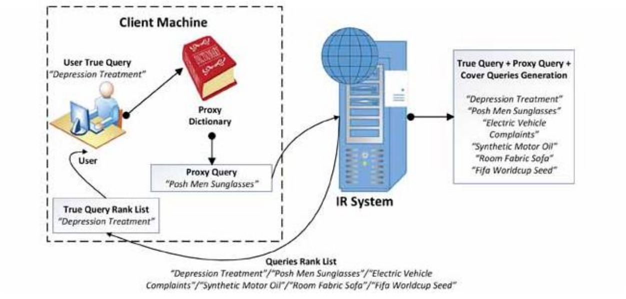
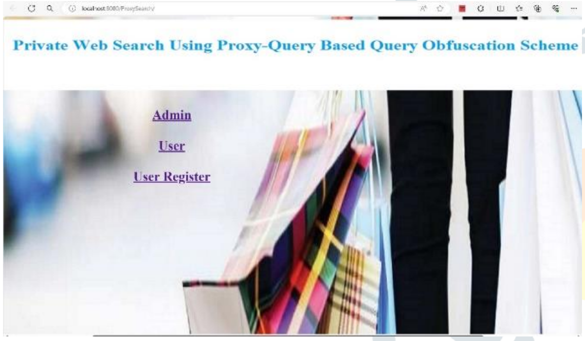


FIG: System Architecture Model

SNAPSHOTS

It implements like web application using COREJAVA and the Server process is maintained using the SOCKET & SERVERSOCKET and the Design part is played by Cascading Style Sheet.

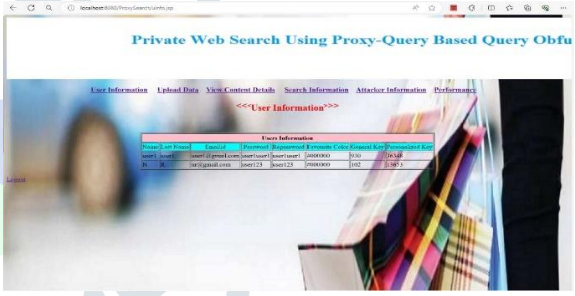


Private Web Search Using Proxy-Query Based Query Obfuscation Scheme

Admin

User

User Register



Private Web Search Using Proxy-Query Based Query Obfu

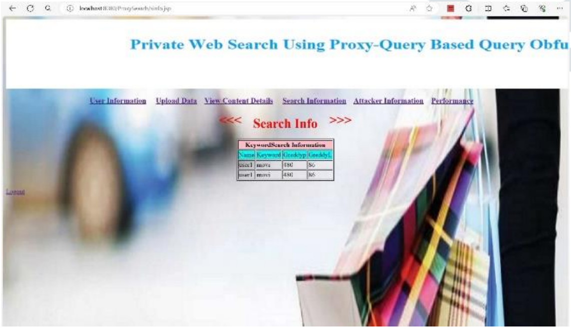
User Information Upload Data View Content Details Search Information Attacker Information Performance

<<<User Information>>>

Logout

Users Information							
Name	Last Name	Emailid	Password	Repassword	Favourite Color	General Key	Personalized Key
user1	user1	user1@gmail.com	user1user1	user1user1	#000000	930	36348
N	R	nr@gmail.com	user123	user123	#000000	102	13653

FIG : Admin Page **FIG: User Information**



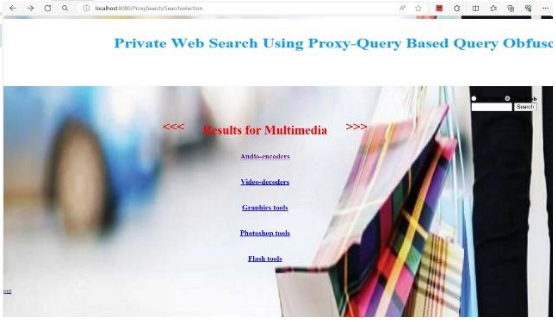
Private Web Search Using Proxy-Query Based Query Obfu

User Information Upload Data View Content Details Search Information Attacker Information Performance

<<< Search Info >>>

Logout

KeywordSearch Information			
Name	Keyword	Gradelyp	Gradelyl
user1	movi	480	86
user1	movi	480	86



Private Web Search Using Proxy-Query Based Query Obfusc

<<< Results for Multimedia >>>

Audio-encoders

Video-decoders

Graphics tools

Photoshop tools

Flash tools

FIG: Search Info **FIG: Result for Multimedia**

V. DEVELOPING METHODOLOGIES

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used. The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies[10].

TYPES OF TESTS

1. Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a [11] structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

2. Functional Test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked[12].

3. System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

4. Performance Test

The Performance test ensures that the output be produced within the time limits, and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

5. Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.[13]The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

6. Acceptance Testing for Data Synchronization:

- The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node
- The Route add operation is done only when there is a Route request in need
- The Status of Nodes information is done automatically in the Cache Updation process

7. Build the test plan

Any project can be divided into units that can be further performed for detailed processing. Then a testing strategy for each of this unit is carried out. Unit testing helps to identity the possible bugs in the individual component, so the component that has bugs can be identified and can be rectified from errors[14].

TEST CASES:

Test cases can be divided in to two types. [15]First one is Positive test cases and second one is negative test cases. In positive test cases are conducted by the developer intention is to get the output. In negative test cases are conducted by the developer intention is to don't get the output.

TEST PLAN

The test procedure is started by building up a thorough arrangement to test the general usefulness and extraordinary highlights on an assortment of stage mixes. Exacting quality control methods are utilized. The procedure checks that the application meets the necessities indicated in the framework prerequisites report and is sans bug[16].

VI. CONCLUSION

Ensuring web search privacy is paramount for internet users. In our study, we propose a privacy scheme employing proxy-query-based query obfuscation to safeguard users' web search privacy. This innovative approach, still in its infancy within the private web search (PWS) domain, facilitates information retrieval from search engines through proxy queries. A notable advantage of our method is that users avoid directly issuing true queries to search engines, enhancing privacy.

We address a limitation of existing proxy-term-based query obfuscation methods, particularly their effectiveness when queries involve multiple terms. The computational challenge of optimizing proxy-term mapping for various query term combinations restricts their efficacy. To overcome this hurdle, our scheme focuses on mapping queries of entire topics rather than individual terms. This strategy significantly improves effectiveness for both single queries and sequences of related queries.

In the latter part of our study, we delineate key characteristics essential for private web search, providing a practical evaluation framework for PWS schemes. These defined characteristics serve as a benchmark for users and researchers to assess the efficacy of existing or newly developed PWS schemes objectively. Through analysis, we find that none of the evaluated PWS schemes meet all the defined characteristics, leaving them susceptible to web search privacy breaches and undermining user trust in web search privacy protection.

VII. REFERENCES

[1] J. Teevan, S. T. Dumais and E. Horvitz, "Personalizing search via automated analysis of interests and activities", ACM SIGIR Forum, vol. 51, no. 3, pp. 10-17, Feb. 2017.

[2] B. Chor, N. Gilboa and M. Naor, "Private information retrieval by keywords", IACR Cryptol. ePrint Arch., vol. 1998, pp. 3, Feb. 1998.

[3] E. Kushilevitz and R. Ostrovsky, "Replication is not needed: Single database computationally-private information retrieval", Proc. 38th Annu. Symp. Found. Comput. Sci., pp. 364-373, 1997.

[4] E. Balsa, C. Troncoso and C. Díaz, "OB-PWS: Obfuscation-based private web search", Proc. IEEE Symp. Secur. Privacy (SP), pp. 491-505, May 2012.

[5] R. Dingledine, N. Mathewson and P. F. Syverson, Proc. 13th USENIX Secur. Symp., pp. 303-320, Aug. 2004.

[6] H. Corrigan-Gibbs and B. Ford, "Dissent: Accountable anonymous group messaging", Proc. 17th ACM Conf. Comput. Commun. Secur. (CCS), pp. 340-350, 2010.

[7] S. B. Mokhtar, G. Berthou, A. Diarra, V. Quema and A. Shoker, "RAC: A freerider-resilient scalable anonymous communication protocol", Proc. IEEE 33rd Int. Conf. Distrib. Comput. Syst., pp. 520-529, Jul. 2013.

[8] M. Ullah, R. Khan, I. U. Khan, N. Aslam, S. S. Aljameel, M. I. U. Haq, et al., "Profile aware obscure logging (paoslo): A web search privacy-preserving protocol to mitigate digital traces", Secur. Commun. Netw., vol. 2022, pp. 2109024:1-2109024:13, 2022.

[9] J. Domingo-Ferrer, M. Bras-Amorós, Q. Wu and J. Manjón, "User-private information retrieval based on a peer-to-peer community", Data Knowl. Eng., vol. 68, no. 11, pp. 1237-1252, Nov. 2009.

[10] K. Stokes and M. Bras-Amorós, "On query self-submission in peer-to-peer user-private information retrieval", Proc. 4th Int. Workshop Privacy Anonymity Inf. Soc. (PAIS), pp. 7, 2011.

[11] A. Arampatzis, P. S. Efraimidis and G. Drosatos, "A query scrambler for search privacy on the internet", Inf. Retr., vol. 16, no. 6, pp. 657-679, Dec. 2013.

[12] A. Arampatzis, G. Drosatos and P. S. Efraimidis, "Versatile query scrambling for private web search", Inf. Retr. J., vol. 18, no. 4, pp. 331-358, Aug. 2015.

[13] C. Wei, Q. Gu, S. Ji, W. Chen, Z. Wang and R. Beyah, "OB-WSPES: A uniform evaluation system for obfuscation-based web search privacy", IEEE Trans. Dependable Secure Comput., vol. 18, no. 6, pp. 2719-2735, Dec. 2019.

[14] P. Yu, W. U. Ahmad and H. Wang, "Hide-n-seek: An intent-aware privacy protection plugin for personalized web search", Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., pp. 1333-1336, Jun. 2018.

[15] C. D. Howe and H. Nissenbaum, "TrackMeNot: Resisting surveillance in web search" in Lessons from the Identity Trail: Anonymity Privacy and Identity in a Networked Society, Oxford, U.K.:Oxford Univ. Press, 2009.

[16] W. U. Ahmad, M. M. Rahman and H. Wang, "Topic model based privacy protection in personalized web search", Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., pp. 1025-1028, Jul. 2016.