2012 IEEE Symposium on Security and Privacy

OB-PWS: Obfuscation-Based Private Web Search

Ero Balsa, Carmela Troncoso and Claudia Diaz
ESAT/COSIC, IBBT
KU Leuven
Leuven, Belgium
Email: firstname.secondname@esat.kuleuven.be

***Abstract*—**
Obfuscation-based private web search (OB-PWS) solutions allow users to search for information in the Internet while concealing their interests. The basic privacy mechanism in OB-PWS is the automatic generation of dummy queries that are sent to the search engine along with users' real requests. These dummy queries prevent the accurate inference of search profiles and provide query deniability. In this paper we propose an abstract model and an associated analysis framework to systematically evaluate the privacy protection offered by OB-PWS systems. We analyze six existing OB-PWS solutions using our framework and uncover vulnerabilities in their designs. Based on these results, we elicit a set of features that must be taken into account when analyzing the security of OB-PWS designs to avoid falling into the same pitfalls as previous proposals.

I. Introduction

Web search has become a regular activity in our lives, as it is often the fastest and most effective way of finding information. Web search service providers, commonly known as search engines, maintain a database of pointers to pages in the Web. These pointers are indexed by keywords, which relate to the content of the associated pages. In order to perform a search in this database, a user composes a query formed by one or more keywords related to the topics she is interested in, and sends it to the search engine. The engine, based on the keywords contained in the query, compiles a list of web pages likely to contain the information of interest and returns it to the user.

Search queries are closely related to the issues we are interested or concerned about, and are thus a rich source to perform user profiling. This raises privacy concerns with respect to social sorting and discrimination, particularly as potentially sensitive information can be inferred from search queries, such as income level, health issues, or political beliefs [19], [29].

Different approaches can be taken to address this problem. Users may connect to the search engine through an anonymous web browsing system [3], [8], [26], which makes them appear as having a different identity in each session; or they may be identifiable but conceal their search profile. We note that these two approaches are complementary. Anonymizers hinder the creation of search profiles through query unlinkability; while concealing the search profile makes it harder to re-identify anonymous users through their queries.

Private information retrieval (PIR) [16], [21] is a class of solutions to conceal search queries. PIR allows a user to retrieve a record from a database without the database owner being able to determine which record was accessed, and PIR schemes have also been proposed in the context of web search [4]. These cryptography-based solutions provide strong privacy guarantees, but require the search engine to implement and run the protocols. Search engines however do not have any incentives to implement costly protocols they cannot profit from, and thus the deployment of these solutions may not be realistic in practice.

In this paper we focus on a category of private web search solutions that we call obfuscation-based private web search (OB-PWS) systems [9], [11], [12], [13], [14], [18], [20], [22], [23], [25], [28], [30]. One of the main advantages of OB-PWS over PIR solutions is that they do not require the cooperation of the search engine. The basic OB-PWS mechanism consists in automatically generating dummy (fake) search queries. These dummy queries, generated by an OB-PWS tool (e.g., a browser plugin), are not necessarily relate to the actual interests of the user. As a result, dummy queries introduce "noise" in the user profile obtained by the search engine, enabling the concealment of her actual interests. Furthermore, if confronted with a sensitive or uncomfortable query, users may claim that it was generated by the OB-PWS tool and obtain plausible deniability about having issued the query.

We note that besides protecting individual users, obfuscation diminishes the overall utility of search profiles to search engines and, assuming that a sufficiently large user base adopts OB-PWS solutions, it may reduce the economic incentives to perform mass sophisticated profiling.

The contributions of this paper are the following:

- We propose an abstract model that captures the key elements of OB-PWS systems and models the capabilities of a strategic adversary.
- We describe an evaluation framework for OB-PWS strategies. We define privacy properties for both search profiles and individual queries, point out the elements that must be considered in the security analysis, and propose metrics to evaluate the effectiveness of different dummy generation strategies.

DOI 10.1109/SP.2012.36

- Based on our model and evaluation framework, we evaluate six proposed OB-PWS systems and uncover vulnerabilities in their designs as well as flaws in their original evaluations.
- We identify key features in OB-PWS systems and discuss their impact on the system properties.
- We provide an overview of open problems and challenges that need to be addressed in order to design effective and robust OB-PWS tools.

II. An Abstract Model for Obfuscation-Based Private Web Search (OB-PWS) Systems

We consider a model in which a user Alice queries a web search engine to find information in the web. Alice's *queries* consist of a set of *keywords* that are related to the information she is looking for. Keywords are processed by the search engine in order to find relevant web pages and return them to Alice. We assume that Alice does not connect to the search engine through an anonymous communication channel [3], [8], [26], and thus consider that her queries can be linked together.

Alice's queries can be associated to *topics* or *categories* according to the keywords in the query and other contextual information. Alice's *search profile* is modeled as a multinomial distribution $X = \{x_i\}$ that we call *real profile*. Each element x_i of Alice's profile represents her level of interest in category or topic i. Usually, x_i is computed as the fraction of queries containing keywords related to category i, according to some *semantic classification algorithm* (SCA).

We note that modeling the profiles as multinomial distributions does not impose constraints on the semantic classification algorithm SCA that associates queries to categories. Categories may range from very broad (e.g., health, sports, music) to very specific, to the extreme of considering each individual keyword as a category.

The OB-PWS adversary is an honest-but-curious search engine, or any other entity with access to the user search queries (e.g., an eavesdropper). The goal of the adversary is to infer private information about Alice from her search profile and queries. For this, the adversary records all the queries received from Alice, and builds an *observed profile* $Y = \{y_i\}$. When all the queries received are real queries issued by Alice herself, Y accurately represents Alice's real profile X (i.e., Y=X).

An OB-PWS tool is a piece of software (e.g., a browser plugin) that runs in Alice's computer. This tool generates *dummy queries*, denoted as D, that are submitted along with Alice's *real queries*, denoted as R. Dummy queries are fake queries that are automatically generated by the OB-PWS tool, and thus are not necessarily related to Alice's real interests. Dummy queries mitigate the privacy threats derived from search profiling by obfuscating the observed profile Y, which now contains a mix of real and dummy queries (i.e., $Y \neq X$). Without loss of generality our model abstracts dummy keywords attached to user queries as separate queries sent simultaneously (e.g., the query "real OR dummy" is modeled as two queries "real" and "dummy").

The OB-PWS tool generates dummy queries according to a *dummy generation strategy* DGS. Typically, the DGS uses a semantic classification algorithm SCA_{DGS} that provides a mapping between the queries and the categories associated with them. The DGS establishes the ratio of dummy queries to be generated, their content and semantics, their distribution amongst categories, the metadata associated to them, the time when they are issued, and any other feature relevant for the operation of the OB-PWS tool.

In order to be effective, dummy queries need to be indistinguishable from real queries. Otherwise the adversary may be able to filter them out and recover a *filtered profile* $Z = \{z_i\}$ that is similar to the real profile X – thus neutralizing the effect of the OB-PWS tool. Similarly, if the DGS distorts the observed profile Y in a way that is predictable and invertible, the adversary can remove (part of) the noise and obtain a filtered profile Z that is a less noisy version of X than Y.

We consider that the filtering of Y to obtain Z combines two algorithms. The first is the *dummy classification algorithm* (DCA). The function of the DCA is to classify queries as either real Q_R or dummy Q_D, based on relevant features of the dummy generation strategy, such as query semantics, grammar, timing, or metadata. When constructing the filtered profile Z, the adversary discards queries Q_D classified as dummies and only takes into account queries Q_R classified as real. The DCA fully succeeds in filtering dummy queries when all queries D and R are correctly classified as Q_D and Q_R, respectively. If the classification of a query as Q_R or Q_D is independent of the query actually being real or dummy, then we say that the DCA fails to provide any useful information to the adversary.

The second component is the *profile filtering algorithm* (PFA). This algorithm attempts to predict the way in which the dummy queries added by the DGS modify each of the components of Alice's real profile, and then invert their effect to recover a filtered profile $Z = \{z_i\}$ that better represents the actual interests of the user. The PFA fully succeeds when the filtered profile Z does not contain any noise (i.e., $Z = X$).

Note that the DCA and PFA algorithms benefit from each other: more information about the real profile X helps identifying dummy queries, and vice versa. We assume the adversary takes advantage of this and runs the algorithms iteratively, refining the filtering.

Figure 1 summarizes the elements of the model. From left to right the figure displays a user issuing *real queries* R which can be represented (according to some SCA) as a profile X. The OB-PWS tool installed in the user's computer receives as input the user's *real queries* R and automatically

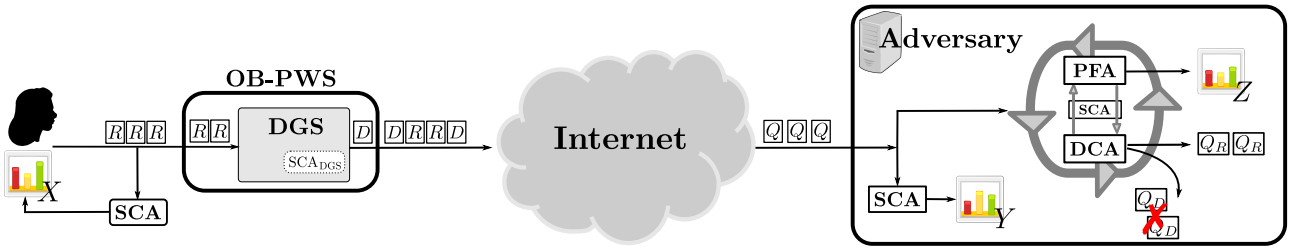

Figure 1. An abstract model for obfuscation-based private web search

generates *dummy queries* D according to its *dummy generation strategy* DGS and associated *semantic classification algorithm* SCA_{DGS}. Both real and dummy queries are sent to the adversarial web search service provider, who (ideally) cannot distinguish them and thus are represented as Q. The observed profile Y is a representation of all Q queries according to some SCA of the adversary's choice. Further, the adversary can implement *dummy classification* DCA and *profile filtering* PFA algorithms that exploit vulnerabilities in the DGS. The former is used to classify queries Q as real Q_R or dummy Q_D, while the latter reverses the obfuscation introduced by the DGS in Y in order to obtain the *filtered profile* Z. The DCA and PFA are applied iteratively (using an SCA to translate queries to semantic categories) to both reduce the amount of noise in Z and enhance the distinguishability of real and dummy queries.

III. Evaluation framework for OB-PWS strategies

In this section we outline an evaluation framework for OB-PWS systems. We define privacy properties for both search profiles and individual queries, point out the elements that must be considered in the analysis, and propose metrics to assess and compare the effectiveness of different dummy generation strategies with respect to the defined privacy properties.

We recall that the query-based and profile-based analyses are complementary, i.e., succesfully identifying dummy and real queries leaks information about the real profile X, and vice versa. A key element connecting both types of analysis is the semantic classification algorithm, SCA. The function of the SCA is to translate query logs into profiles, by associating queries to profile categories.

The evaluation of an OB-PWS dummy generation strategy (DGS) requires exploring the possible adversarial strategies (DCA, SCA, and PFA) and their success in: (1) recovering the user's real profile X; and (2) identifying with a high degree of certainty the user's real queries R.

A. Profile-Based Analysis.

Our profile-based analysis aims to measure the uncertainty of the adversary on Alice's real profile X after it has been obfuscated by the dummy generation strategy DGS.

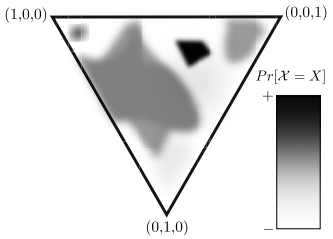

Figure 2. $\Pr[\mathcal{X} = X]$ in the profile space.

Analyzing the level of profile privacy provided by a dummy generation strategy requires exploring semantic classification and profile filtering algorithms that could be implemented by the adversary in order to filter observed profiles and extract as much information as possible about user preferences and interests. The amount of profile information leaked by the DGS is an indicator of the level of protection provided by an OB-PWS design. This is given by the difference between the a priori and a posteriori uncertainty of the adversary on the real profile X, i.e., before and after obtaining the observed Y and filtered Z profiles.

We assume that the adversary has background information on the interests of the user population (e.g., which search topics are more popular). We model this information as a random variable $\mathcal{X}$. $\Pr[\mathcal{X} = X]$ describes the (a priori) probability that a user has a particular profile X, where X is a vector with as many dimensions as categories considered by the SCA. Figure 2 shows an example of the probability density $\Pr[\mathcal{X} = X]$, simplified to three dimensions, i.e., profiles $X = \{x_1, x_2, x_3\}$ that have three components $0 \leq x_i \leq 1$ such that $\sum_i x_1 = 1$. Darker areas represent highly likely profiles, while lighter areas refer to rare profiles. We measure the adversary's (a priori) uncertainty on X as the entropy [27] of $\mathcal{X}$, $H(\mathcal{X})$.

The adversary can construct an observed profile Y with the queries submitted by the user and the OB-PWS tool. Let $\mathcal{Y}$ be a random variable representing the probability of occurrence of observed profiles, and let $\mathcal{E}_{\mathcal{Y}}$ denote the conditional entropy (also known as *equivocation*) of $\mathcal{X}$ given $\mathcal{Y}$:

$$\mathcal{E}_{\mathcal{Y}} = H(\mathcal{X}|\mathcal{Y}) = H(\mathcal{X}, \mathcal{Y}) - H(\mathcal{Y})\,.$$

$\mathcal{E}_{\mathcal{Y}}$ is the average uncertainty of the adversary on real profiles $X \in \mathcal{X}$ given observed profiles $Y \in \mathcal{Y}$. The average amount of information leaked by observed profiles on real profiles is given by $H(\mathcal{X}) - \mathcal{E}_{\mathcal{Y}}$.

After recovering Y, a *strategic adversary* aware of the use of the OB-PWS tool can apply DCA and PFA algorithms to obtain a filtered profile Z. We define $\mathcal{Z}$ and $\mathcal{E}_{\mathcal{Z}}$ analogously to $\mathcal{Y}$ and $\mathcal{E}_{\mathcal{Y}}$:

$$\mathcal{E}_{\mathcal{Z}} = H(\mathcal{X}|\mathcal{Z}) = H(\mathcal{X}, \mathcal{Z}) - H(\mathcal{Z})\,.$$

$\mathcal{E}_{\mathcal{Z}}$ is the average uncertainty of a strategic adversary on real profiles $X \in \mathcal{X}$ given filtered profiles $Z \in \mathcal{Z}$. The average amount of profile information leaked by the DGS on real profiles is given by $H(\mathcal{X}) - \mathcal{E}_{\mathcal{Z}}$.

An OB-PWS system provides *perfect profile protection* when the adversary is unable to gain *any* information about Alice's real profile X from Z; i.e., $\mathcal{E}_{\mathcal{Z}} = H(\mathcal{X})$. Conversely, when $\mathcal{E}_{\mathcal{Z}} = 0$ the information leaked by the DGS is $H(\mathcal{X})$, and the adversary can perfectly reconstruct real profiles $\mathcal{X}$ from filtered profiles $\mathcal{Z}$. Formally, $\forall Z \in \mathcal{Z}$, $\exists X \in \mathcal{X}$ such that $\Pr[\mathcal{X} = X|\mathcal{Z} = Z] = 1$.

In this paper we use $\mathcal{E}_{\mathcal{Z}}$ as a metric to illustrate how previous analyses of OB-PWS tools oversee information leaked by the used DGS hence overestimating the protection provided by these systems. However, we note that $\mathcal{E}_{\mathcal{Z}}$ only gives a measure of the *average* level of protection provided by a dummy generation strategy to user profiles. When $\mathcal{E}_{\mathcal{Z}} < H(\mathcal{X})$, this metric does not give any guarantee on the protection given to specific individual profiles, and further metrics should be taken into account in a comprehensive analysis.

B. Query-Based Analysis

One of the goals of the OB-PWS dummy generation strategy DGS is to issue dummy queries D that are indistinguishable from real queries R. A query-based analysis requires first studying which features of the DGS (e.g., semantics, metadata) could be exploited by a DCA to distinguish between real and dummy queries. *Perfect query protection* is provided when for all possible dummy classification algorithms DCA the probability of a query being classified as Q_R (or Q_D) is independent of the query actually being real R or dummy D; i.e., $\Pr[Q_R|Q = R] = \Pr[Q_R|Q = D]$, and analogously, $\Pr[Q_D|Q = R] = \Pr[Q_D|Q = D]$. Figure 3 shows the probabilities associated with the dummy classification algorithm.

On the other hand, if the adversary can implement a dummy classification algorithm DCA that classifies all queries correctly (i.e., $\Pr[R|Q_R] = \Pr[D|Q_D] = 1$), then the OB-PWS system offers no query privacy protection. Note that this implies that the filtered profile will contain all real queries and no dummies, and thus $Z = X$ and $\mathcal{E}_{\mathcal{Z}} = 0$.

We consider two query-based privacy properties to evaluate the protection offered by a DGS: *unobservability*, denoted as $\mathcal{U}$, and *deniability*, denoted as $\mathcal{D}$. A real query R is *unobservable* when the adversary classifies it as a dummy query Q_D. We recall that queries classified as Q_D are discarded when constructing the filtered profile Z. Thus, unobservable queries hinder the reconstruction of Z by misrepresenting the weight of the categories associated with unobservable queries.

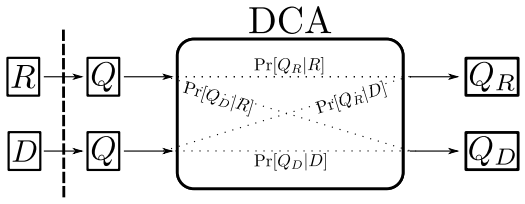

Figure 3. DCA

We define the average level of unobservability ($\mathcal{U}$) provided by a DGS to user queries as the fraction of real queries R that are misclassified as dummies Q_D by the adversary's DCA:

$$\mathcal{U} = \Pr[Q_D|Q = R].$$

Unobservability ranges from $\mathcal{U} = 0$, when all real queries are correctly identified, to $\mathcal{U} = \Pr[D]$, when real queries are misclassified as Q_D at the same rate as the ratio dummy queries to total queries $\Pr[D] = \frac{D}{R+D}$. We assume that for a non-trivial DCA, the adversary never misclassifies more real queries than correctly classifies dummies, i.e., $\Pr[Q_D|Q = R] \leq \Pr[Q_D|Q = D]$.

Even if some (or many) of the users' queries are unobservable, a fraction $\Pr[Q_R|Q = R]$ of real user queries are still classified as real by the adversary and taken into account for the construction of the filtered profile Z. If a significant fraction $\Pr[Q_R|Q = D]$ of dummy queries are also classified as Q_R, the user can plausibly deny having issued a query R, and claim instead that the query was in fact a dummy D generated by the OB-PWS tool. We measure the average level of *deniability* ($\mathcal{D}$) provided by a DGS to user queries as:

$$\mathcal{D} = \Pr[D|Q_R] = \frac{\Pr[Q_R|Q = D] \cdot \Pr[D]}{\Pr[Q_R]}\,.$$

Deniability ranges from $\mathcal{D} = 0$, when no dummy queries are misclassified as Q_R, to $\mathcal{D} = \Pr[D]$, when dummy and real queries are classified as Q_R with the same probability ($\Pr[Q_R|Q = D] = \Pr[Q_R|Q = R]$) and thus the adversary's best guess can only be based on his a-priori information on the proportion of dummy queries issued by the OB-PWS tool.

Table I offers a summary of the notation we have introduced throughout this section.

Table I
SUMMARY OF NOTATION

Symbol	Meaning
R	Real query issued by the user
D	Dummy query issued by the OB-PWS tool
Q	Query (real or dummy) observed by the adversary
$X = \{x_i\}$	Real Profile. Multinomial distribution representing the user's level of interest in different categories according to some SCA
$Y = \{y_i\}$	Observed Profile. Multinomial distribution representing the adversary's view of the interests of the user different categories according to some SCA
$Z = \{z_i\}$	Filtered Profile. Multinomial distribution representing the adversary's view of the interests of the user according to some SCA after applying DCA and PFA algorithms
Q_R	Query (R or D) that the adversary classifies as real
Q_D	Query (R or D) that the adversary classifies as dummy
$\mathcal{D}$	Deniability
$\mathcal{U}$	Unobservability
DGS	*Dummy generation strategy* of the OB-PWS system
SCA	*Semantic classification algorithm* that associates queries to the categories considered in the profile
DCA	*Dummy classification algorithm* implemented by the adversary that exploits weaknesses in the DGS to classify queries as either Q_R or Q_D
PFA	*Profile filtering algorithm* implemented by the adversary that exploits weaknesses in the DGS to predict the noise added by the DGS to X in order to filter it out of Y
$\mathcal{X}$	Random variable describing the probability over all possible real profiles X
$\mathcal{Y}$	Random variable describing the probability over all possible observed profiles Y
$\mathcal{Z}$	Random variable describing the probability over all possible filtered profiles Z
$\mathcal{E}_\bullet$	Equivocation or conditional entropy representing the average uncertainty of the adversary on real profiles X given profiles $\bullet$, $\bullet = \{Y, Z\}$

IV. OBFUSCATION-BASED PRIVATE WEB SEARCH

In this section we review six OB-PWS systems that have been proposed in the literature. We consider that these papers, which implement various different strategies, are a good representation of the state-of-the-art in obfuscation-based private web search.

A. TrackMeNot: Resisting Surveillance in Web Search

TrackMeNot (TMN) is a popular[1] browser plugin designed by Howe and Nissembaum [18]. TMN generates dummy queries, D, that are sent together with Alice's real queries, R, in order to introduce noise in the observation of the adversary and prevent the recovery of Alice's search profile X. TMN implements a number of strategies to generate dummy queries. Although TMN focuses mainly on generating plausible dummy queries, it seeks profile privacy protection (informally defined as dissimilarity between the real and observed profiles) rather than query deniability. TMN does not formally define privacy properties and its security is not evaluated against an adversary that is aware of the plugin and tries to neutralize its effect [18].

TMN has been found to be vulnerable to DCA attacks that exploit the semantics [2] and grammatical construction [5] of dummy queries to distinguish them from real queries. Naïve machine learning techniques [24] have also been shown to be effective in distinguishing dummy queries, assuming that a sample of Alice's browsing history (i.e., real queries) is available for training the algorithms.

There are a number of other features in the DGS of TMN that could be exploited by a DCA to identify and filter out dummy queries. In TMN, dummy queries are composed by keywords drawn from a "Dynamic Query List" [2] initialized with a list of common query terms extracted from: i) RSS feeds from popular websites such as Slashdot or CNN, and ii) a list of popular search terms (e.g., extracted from Google Trends[2]).

The initialization sources of the Dynamic Query List are public. Let "popular" refer to keywords that appear frequently in the Dynamic Query List. A query $Q_{\overline{\text{popular}}}$ that does not contain any "popular" keywords, can be thus classified as Q_R, and enjoys a low level of unobservability; i.e., $\Pr[Q_D|R = Q_{\overline{\text{popular}}}] \approx 0$. Note that these queries are not deniable either, as Alice cannot plausibly claim that the OB-PWS tool generated a query $Q_{\overline{\text{popular}}}$; i.e., $\Pr[D|Q_R = Q_{\overline{\text{popular}}}] \approx 0$.

TMN updates the Dynamic Query List with keywords from Alice's real queries, so that future dummy queries are plausible and concordant with her search history. While this strategy enhances individual query unobservability and deniability, it also reduces profile obfuscation, as dummy queries are distributed in categories similarly to real queries. Therefore, even if some dummy queries are misclassified as real, they will only introduce small amounts of noise in the filtered profile – ultimately defeating TMN's goal of obfuscating user interests and preferences.

TMN also specifies techniques for constructing the metadata of dummy queries. The reuse of real queries' metadata in dummy queries makes the tool vulnerable to DCAs that exploit query metadata. "Live Header Maps" ensure that dummy requests generated by TMN have as headers the *last* set of headers issued by the browser. Hence, every time a query $Q_{\text{new headers}}$ with new headers is received, the DCA determines that the query is real, as otherwise the headers would have remained unaltered; i.e., $\Pr[Q_D|R = Q_{\text{new headers}}] \approx 0$), and $\Pr[D|Q_R = Q_{\text{new headers}}] \approx 0$. In other words, real queries containing new values in the header are observable and undeniable.

Finally, TMN implements a "Cookie Anonymization" mechanism that mandates that cookies are *only* sent with dummy queries. TMN assumes that queries sent *without*

[1]As of March 2012, Mozilla reports more than 42 000 users of TMN (https://addons.mozilla.org/en-US/firefox/addon/trackmenot/).

[2]http://www.google.com/trends

cookie ($Q_{\overline{\text{cookie}}}$) are anonymous, and not linkable to queries sent *with* cookie (Q_{cookie}). However, it has been shown that browser fingerprinting techniques can be used to trivially link together all the queries sent by a browser [10]. Thus, the adversary can exploit the presence or absence of a cookie as an indicator of whether the query is real or dummy (i.e., $\Pr[Q_D|R = Q_{\overline{\text{cookie}}}] \approx 0$, and $\Pr[D|Q_R = Q_{\overline{\text{cookie}}}] \approx 0$).

The various exploitable features of TMN's dummy generation strategy reviewed in this section enable an adversary to implement a DCA that classifies queries correctly with high probability. Distinguishing and filtering out dummy queries helps the adversary refine the filtered profile Z, so that it is an accurate reconstruction of Alice's real profile X.

B. GooPIR: $h(k)$-Private Information Retrieval from Privacy-Uncooperative Queryable Databases

GooPIR[3] [9], similarly to TMN, selects keywords from a public dictionary to construct dummy queries. For each of Alice's real queries R, GooPIR generates $k-1$ dummy queries D, which are submitted together with R. The simultaneous submission of real and dummy queries prevents the adversary from exploiting query timing or metadata to identify dummies. On the other hand this strategy does not conceal *when* Alice is submitting a real query. Although sequences of real query timings may potentially be exploitable by an adversary, DCA algorithms that consider this information are not explored in this paper and are left as subject for future work.

GooPIR aims to offer what Domingo-Ferrer et al. call $h(k)$-*private information retrieval* ($h(k)$-PIR). This property ensures that a real query R is seen by the adversary as a random variable $\mathcal{R}$ whose entropy is such that $H(\mathcal{R}) \geq h(k)$ for some function h. GooPIR describes a protocol to construct dummy queries such that they are perfectly indistinguishable from the real queries (i.e., such that $H(\mathcal{R}) = \log(k)$). When perfect indistinguishability is achieved, each of the k queries Q is classified as dummy with probability $\Pr[Q_D|Q] = \frac{k-1}{k}$, and as real with probability $\Pr[Q_R|Q] = \frac{1}{k}$.

GooPIR seeks a compromise between computational efficiency and privacy. Domingo-Ferrer et al. argue that the higher k, the more dummy queries are sent to the search engine, and the more privacy the system offers. In terms of our query-based privacy properties, achieving $\log(k)$-*PIR* corresponds to maximum query unobservability and deniability ($\mathcal{U} = \mathcal{D} = \Pr[D] = \frac{k-1}{k}$), which tend to one as k increases.

Domingo-Ferrer et al. point out that the adversary may be able to use a DCA that exploits the "popularity" of queries (as explained for TMN) to identify and remove dummies. To counter this attack GooPIR checks the popularity of the keywords in the real query, and selects keywords for the $k-1$ dummy queries that have a similar level of popularity. GooPIR assumes that the "popularity" of a query is proportional to its frequency of appearance in the Web, and that a public dictionary labeled with such frequencies is available.

[3] http://unescoprivacychair.urv.cat/goopir.php

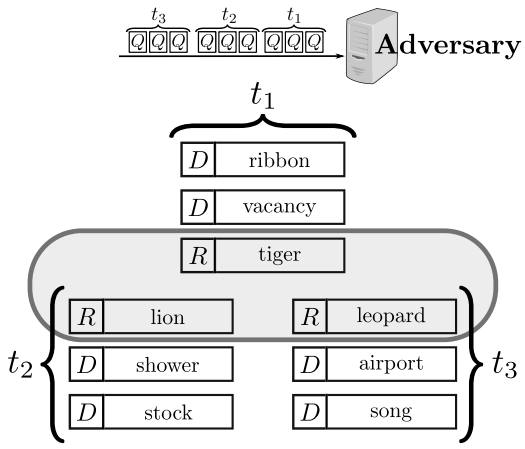

Figure 4. SCA attack on GooPIR

Further, to prevent disclosure attacks [1], [7] a query R is *always* accompanied by the *same set* of $k-1$ queries D. By accompanying real queries always with the same set of dummy queries, GooPIR prevents real queries from appearing more frequently than dummies.

Domingo-Ferrer et al. provide in [9] a query-based analysis of GooPIR in which they evaluate the distinguishability of real and dummy queries, and conclude that their strategy indeed provides $h(k)$-PIR. Their analysis, however, considers a single set of k queries, and does not take into account that the adversary may combine multiple sets of queries and use a SCA to find correlations in the topics associated with the queries.

To illustrate this, let us consider that $k = 3$ and that Alice has consecutively issued the three sets of queries shown in Fig. 4: {"ribbon", "vacancy", "tiger"}, {"lion", "shower", "stock"}, {"leopard", "airport", "song"}. A SCA may reveal that big cats appear more often than others (see Fig. 4, dark circle), and thus that it is more likely that the user issued the queries {"tiger", 'lion", "leopard"} than any other combination. This implies that GooPIR does not provide the promised perfect query indistinguishability [9] when various sets of queries are taken into account, and consequently, the unobservability and deniability provided to queries also falls below $\frac{k-1}{k}$.

C. Plausibly Deniable Search.

Murugesan and Clifton propose "Plausibly Deniable Search" (PDS) [22], [23], a dummy generation strategy that aims at providing a user with "plausible deniability" with respect to her queries. Analogously to GooPIR, each real query is accompanied by $k-1$ dummy queries, and thus

query timing and metadata cannot be used to distinguish dummy queries. Further, PDS substitutes user queries by *canonical queries* [22], [23] to prevent the identifiability of real queries based, e.g., on grammar or typos. Canonical queries are formed by generic terms that can be combined to represent any topic that could be searched by the users.

Let S denote the set of k queries $S = \{Q_1, \ldots, Q_k\}$, of which one query is real and $k-1$ are dummies. The DGS for choosing the $k-1$ dummy queries follows three rules: (i) any real query $Q_i = R$ must generate the set S with equal probability (i.e., the set S does not leak information about the real query R that generated it); (ii) all Q_i in S relate to different topics (i.e., the set S is diverse with respect to semantic categories); and (iii) all Q_i in S are equally plausible (i.e., no query in S can be filtered out because it is more likely to have been generated by the OB-PWS tool than by a user).

Murugesan and Clifton argue that query sets S constructed following the aforementioned rules provide privacy, as they enable the user to deny having issued $Q_i = R$ and to claim instead that $Q_i = D$ and that her query was a different $Q_j = R$. The reasoning is that this is plausible because any of the k queries is equally likely of having been generated by the user, and they would all result in the same observed set S. Assuming that the three rules are satisfied and that there is no DCA that could identify some queries as being more likely real than others, PDS's definition of "plausible deniability" is equivalent to $\mathcal{D}$ (as defined in Sect. III-B) when maximum deniability and unobservability are achieved ($\mathcal{D} = \mathcal{U} = \Pr[D] = \frac{k-1}{k}$).

To ensure topic diversity, the dummy generation strategy of PDS relies on a SCA_{PDS} called "Query-Topic Score" (denoted as *rscore*). For each query Q, *rscore* computes a vector with as many components as semantic categories are considered by the SCA_{PDS}. The value of each component of the vector is a score that expresses the extent to which Q relates to category i. PDS assumes that a suitable *rscore* algorithm is available, and makes abstraction of its specific implementation. PDS uses the *rscore* vectors to select dummy queries that relate to semantically distant categories, according to a *topic dissimilarity metric* (e.g. cosine similarity).

The experimental evaluation of PDS presented in [22] shows that it generates query sets S that relate to diverse topics. Murugesan and Clifton argue that "the *existence* of k diverse query mappings to the same query set S is sufficient" for obfuscating the user profile X. Their evaluation however falls short of analyzing to what extent a strategic adversary (that considers sequences of queries and background information) would be uncertain with respect to the topics of interest for the user.

To ensure that all queries in S are equally plausible, PDS requires that all k queries $Q_i \in S$ have a similar level of "specificity" with respect to their "dominating topic"; i.e., the maximum value in their respective *rscore* vectors should be comparable. Note that this assumes that "specificity" of queries is the only feature that can be exploited by the DCA to distinguish dummy queries, and disregards other characteristics such as the frequency of appearance of keywords in the Web (which is considered by GooPIR [9]). PDS does however not provide evidence proving that "specificity" is indeed the only (or even most relevant) feature to be considered when analyzing the robustness of its DGS to DCAs.

Given a concrete SCA_{PDS} and a function *rscore*, PDS ensures that two queries R_1 and R_2 that are semantically close generate sets of dummy queries that are also semantically dependent. This aims at preventing attacks, as the one described in the previous section for GooPIR, that exploit correlations in the semantics of the queries in a sequence to identify the real queries. Note however that this implicitly assumes that the adversary will use SCA_{PDS} in her analysis. If the adversary uses a different SCA_{Adv}, the semantic correlation of dummy queries may be weakened compared to that of the real queries, enabling the distinguishability of real queries.

To illustrate this, let us consider a PDS system with $k = 2$ (i.e., each real query is accompanied by one dummy query). Consider for instance a user that issues the queries {"Justin Bieber", "Toy Story", "Disneyland"}, and that according to SCA_{PDS} the dominant topics of these queries are "music", "cartoons", and "amusement parks", respectively.

Further, consider that these categories are always masked by dummy queries about "history", "physics", and "cars", respectively, also according to SCA_{PDS}. Now consider that the adversary implements a different SCA_{Adv} that classifies all three queries "Justin Bieber", "Toy Story", and "Disneyland" as being related to "kids", rather than being associated to "music", "cartoons", and "amusement parks". Given this SCA_{Adv}, it would be apparent to the adversary that topics related to kids appear more often than others, and hence that kid-related queries are likely to be the user's real sequence of queries.

D. PRAW - A PRivAcy model for the Web.

PRAW is an OB-PWS tool which has been proposed, analyzed, and improved in several articles [11], [12], [13], [14], [20], [28]. PRAW generates dummy web transactions to conceal the profile of interests of a user. This profile X (called "Internal User Profile" in PRAW) is computed using a SCA_{PRAW} called "Browser Monitor". The SCA_{PRAW} maps transactions (queries or visited web pages) to a vector that indicates the "weight" of the transaction with respect to each of the considered semantic categories. These vectors are then used to: (1) construct a user profile X that represents her overall interest in the different semantic categories or topics; (2) assess the level of protection that PRAW is providing to X; and (3) feed and trigger the DGS.

PRAW generates (on average) T_r dummy queries for each user real query. The DGS of PRAW constructs dummy queries with *"a mix of terms, originating in the IUP ["Internal User Profile"], along with random terms originating from an internal database of terms that is a glossary of terms related to the general domain of the user's interests"* [14] (where IUP corresponds to X). The goal of this strategy is to generate dummy queries that relate to topics that are not too different from those of the user, and thus prevent the adversary from deploying clustering attacks [14] that distinguish real and dummy queries based on their topic. The authors of PRAW acknowledge that such a strategy may reveal users' broader interests, but argue that it is necessary to generate plausible dummy queries and that preventing the adversary from inferring specific topics of interest offers sufficient privacy protection. For instance, the adversary may discover that a user is interested in computer security, but cannot learn whether her specific interest is cryptography or intrusion detection systems.

PRAW measures profile privacy as the distance between the real and the observed profiles ($S(X,Y)$), computed as the cosine similarity between the vectors X and Y [11], [12], [13], [14], [20], [28]. PRAW considers that the closer $S(X,Y)$ is to zero, the less information Y leaks about X. Accordingly, the DGS of PRAW (called "Transaction Generator") attempts to generate dummies that decrease the similarity $S(X,Y)$.

PRAW has been evaluated against the aforementioned clustering attack [14]. The evaluation found that dummy queries are hard to filter based on their topic, and that the attack results in $S(X,Y)$ that are reasonably low – thus concluding that PRAW provides an adequate level of privacy protection to user profiles X.

The privacy metric used in PRAW implicitly assumes that the cosine similarity between real and observed profiles $S(X,Y)$ is indicative of the uncertainty of the adversary on X. We note that the results reported in [11] indicate that PRAW's strategy works in such a way that the similarity $S(X,Y)$ is a function of the dummy generation rate T_r (e.g., generating 10 dummies per real query results in similarities around 0.7), which can be inferred from the total number of queries generated [24]. We argue that this is not the case, and that a DGS that results in a predictable $S(X,Y)$ can actually be exploited by a PFA to significantly reduce the uncertainty of the adversary on X.

Let us illustrate with a simple example how a PFA can exploit the predictability in PRAW's strategy with respect to the distance between X and Y.

We first consider an adversary who does not have any prior information on the distribution of user profiles $\mathcal{X}$ (i.e., all possible profiles $X \in \mathcal{X}$ are equally likely, and the a priori uncertainty is $H(\mathcal{X}) = \log(|\mathcal{X}|)$). Figure 5 shows the space $\mathcal{X}$ of possible profiles X when considering three categories (vectors $X = \{x_1, x_2, x_3\}$ are such that $\sum_i x_i = 1$). Consider that the adversary observes profile Y, which in the figure corresponds to the point marked as $\bullet$. We denote as $\hat{d}$ the estimated expected value of $1 - S(X,Y)$ given T_r. Given PRAW's strategy, the real profile X that resulted in observation Y lies with high probability in the curve defined by points at distance $\hat{d}$ from Y. In Fig. 5, higher probability densities $\Pr[X|Y,\hat{d}]$ are depicted in a darker shade. The width of the curve is given by the confidence interval of $\hat{d}$. PRAW's strategy leaks that profiles lying in these dark areas are the most likely candidates for being the real profile X of the user – thus significantly reducing the adversary's uncertainty with respect to X (i.e., $\mathcal{E}_{\mathcal{Z}} << H(\mathcal{X})$).

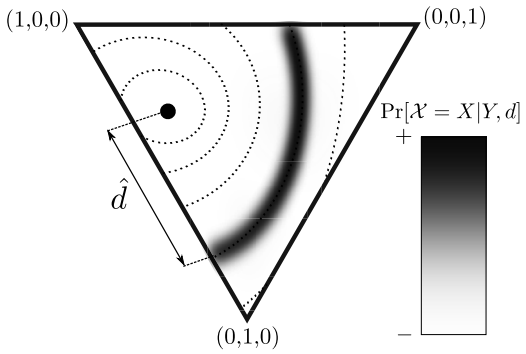

Figure 5. $\Pr[X|Y,\hat{d}]$ assuming that $\Pr[\mathcal{X} = X]$ is uniform (or not available).

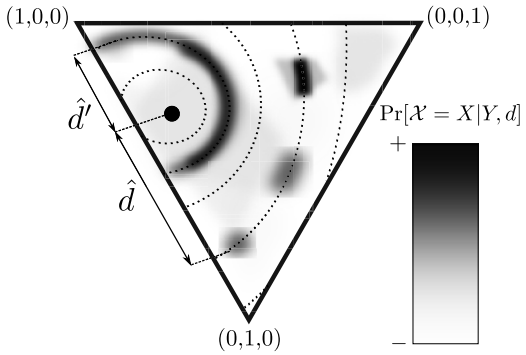

Figure 6. $\Pr[X|Y,\hat{d}]$ and $\Pr[X'|Y,\hat{d}']$ assuming that $\Pr[\mathcal{X} = X]$ is as depicted in Fig. 2 and available to the adversary.

This information leakage is aggravated if the adversary has prior information on which are the likely user profiles X. Let us consider that the prior probability distribution of $\mathcal{X}$, $\Pr[\mathcal{X} = X]$, is for instance as shown in Fig. 2. Bayes' theorem can be used to compute the posterior probability $\Pr[\mathcal{X}|\mathcal{Y},\hat{d}]$. This would help the adversary to further narrow down the set of highly likely profiles to those X that are both reasonably common in the population *and* that lie at a distance $d \approx \hat{d}$ from the observed profile Y. We show in Fig. 6 an example of combining an observation Y with the background information on $\mathcal{X}$, given two possible estimated

distances $\hat{d}$ and $\hat{d}'$.

PRAW considers that privacy is proportional to distance (inversely proportional to similarity), and thus that if $\hat{d}' < \hat{d}$ (conversely $\hat{S}' > \hat{S}$), then the DGS resulting in $\hat{d}$ provides better privacy than the DGS$'$ resulting in $\hat{d}'$. We note that in the scenario depicted in Fig. 6, considering background information may result in $\hat{d}'$ corresponding to a higher level of uncertainty on X (larger dark surface) than $\hat{d}$; i.e., $\mathcal{E}'_{\mathcal{Z}}$ may be higher than $\mathcal{E}_{\mathcal{Z}}$ even though $\hat{d}' < \hat{d}$. This illustrates that distance is not necessarily proportional to privacy, and that using distance-based metrics can result in a misleading privacy evaluation. Furthermore, crafting the DGS to maximize a particular geometric distance metric can be exploited by the adversary, who can invert the noise added by the OB-PWS tool to reduce her uncertainty on the user profile.

E. Optimized Query Forgery for Private Information Retrieval (OQF-PIR)

Rebollo-Monedero and Forné proposed OQF-PIR [25], an OB-PWS system that aims at optimizing the protection provided to user profiles X when a limited budget of dummy queries is available. OQF-PIR assumes that the *population profile* Y^T, a profile describing the aggregate interests of the whole set of users, is known.

Rebollo-Monedero and Forné claim that *"whenever the user's distribution [profile] differs from the population's, a privacy attacker will have actually gained some information about the user, in contrast to the statistics of the general population"*. They propose to measure profile privacy as the Kullback-Leibler (KL) divergence [6] $d_{KL}(Y||Y^T)$ between the observed profile Y and the population profile Y^T. They interpret $d_{KL}(Y||Y^T)$ as a measure of dissimilarity between the observed and population profiles, and consider that privacy is perfectly protected when $d_{KL}(Y||Y^T) = 0$. Additionally, the adversary is assumed to not be aware of the OQF-PIR tool, and thus to take for granted that Y represents the real profile of the user.

We note that, according to this metric, a user Alice whose profile coincides with the average of the population (i.e., $X = Y^T$) would enjoy perfect privacy protection without the need for any obfuscation tool, implying that privacy protection is only needed for users who "deviate" from the average. The adversary would however be able to perfectly reconstruct Alice's profile X. We argue that profile privacy protection relates to the uncertainty of a strategic adversary on the real user profile X, and not to how "average" or "outlier" a user appears to be with respect to the rest of the population (i.e., being revealed as "average" may also lead to a privacy breach).

The DGS of OQF-PIR is designed to optimally minimize $d_{KL}(Y||Y^T)$. OQF-PIR implicitly assumes that a SCA_{OQF} is available to the DGS that identifies query topics and constructs profiles (vectors) representing the interest of the user in each of the topics (modeled as a multinomial distribution).

In order to find the optimal dummy generation strategy OQF-PIR models the observed profile Y as a weighted function of the real profile X and a dummy profile W:

$$Y = (1-\rho)X + \rho W\,, \tag{1}$$

The dummy profile W is a multinomial distribution in which each element w_i represents the fraction of dummy queries in category i to be generated by the DGS. The weighting factor ρ (called redundancy) is the ratio of dummy to total (real and dummy) queries, and represents the limited budget of dummy queries available. For a given real profile X and rate ρ, the optimal dummy profile W is the one that minimizes $d_{KL}(Y||Y^T)$.

The optimization algorithm works by first ordering the profile categories in such a way that

$$\frac{x_1}{y_1^T} \leq \cdots \leq \frac{x_i}{y_i^T} \leq \cdots \leq \frac{x_n}{y_n^T}\,, \tag{2}$$

and then assigning values to their corresponding w_i in a water-filling fashion. That is, dummies are added starting by the first categories until the budget of dummies is exhausted [15]. Let us consider for simplicity that Y^T is the uniform distribution. Assuming that ρ is such that only the first j out of n categories can be completely filled, the resulting observed profile $Y = \{y_1, \cdots, y_n\}$ satisfies that $y_1 = \cdots = y_j < y_{j+1} \leq \cdots \leq y_n$. Note that, as no dummies are added to the last components, $w_i = 0$ and $y_i = (1-\rho)x_i$ for $i > j+1$.

OQF-PIR assumes a non-strategic adversary who does not attempt to attack the dummy generation strategy. We now evaluate DCAs that identify (some of the) real queries, and PFAs that significantly reduce the uncertainty of the adversary on X.

Let us consider an observed profile Y such that its l last components y_i have bigger values than their corresponding y_i^T (i.e., $y_i^T < y_i$, for $n-l < i \leq n$), and let C denote the set of categories $C = \{i\}_{n-l<i\leq n}$. The water-filling DGS implemented by OQF-PIR does not generate any queries on those l categories —as they would take Y farther from, rather than closer to, the target profile Y^T. From a query analysis perspective, the adversary can implement a DCA that exploits this feature, and identifies as Q_R queries Q_C that are associated with categories included in set C according to SCA_{OQF}. Thus, these queries enjoy no unobservability or deniability, as $\Pr[Q_D|Q = Q_C] \approx 0$ and $\Pr[D|Q_R = Q_C] \approx 0$.

OQF-PIR assumes that the dummy rate ρ is a secret parameter. We note however that a rate $\hat{\rho}$ could be estimated from the overall number of queries and default configuration parameters. Let us assume that the adversary is able to estimate a probability distribution of $\hat{\rho}$. We consider a three-dimensional profile space formed by categories (a, b, c), as

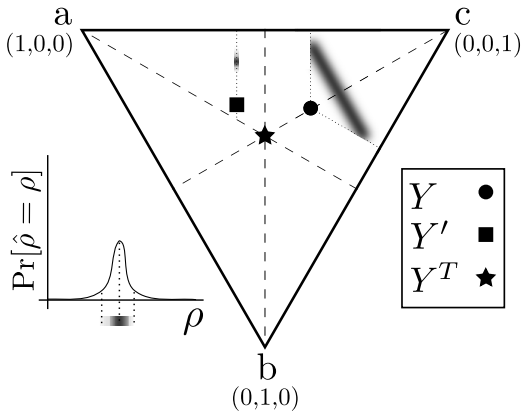

Figure 7. Possible real profiles (as a function of ρ), target profile, observed profile, and implausible real profiles, in the profile space

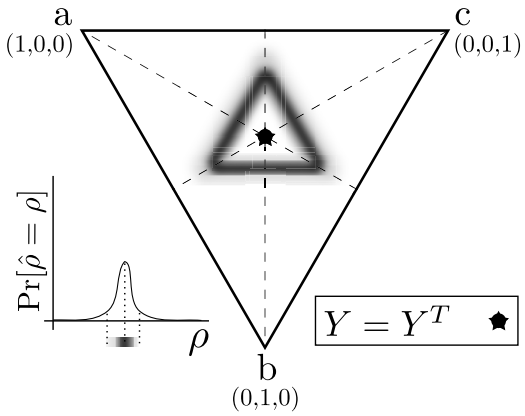

Figure 8. Probability of ρ over a region of the profile space.

shown in Fig. 7, and a population profile that lies at the center of the space; i.e., at point $Y^T = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$.

Given the water-filling algorithm used by the DGS, noise is added to profiles in a deterministic way. Consider that the observed profile is Y', represented as a square dot in Fig. 7. The components of Y' are such that $y'_b < y'_c < y'_a$. The gap between the two smallest components (y'_b and y'_c) indicates that ρ is not sufficient to fill the smallest component (y'_b). The DGS must have generated dummies with a vector $W' = (w'_a, w'_b, w'_c) = (0, 1, 0)$, and thus the real profile X' can be estimated as:

$$\hat{X}' = (\frac{y'_a}{1-\hat{\rho}}, \frac{y'_b - \hat{\rho}}{1-\hat{\rho}}, \frac{y'_c}{1-\hat{\rho}}) .$$

Note that as $\hat{\rho} \to \rho$, $\hat{X}' \to X'$ and $\mathcal{E}_{Z'} \to 0$, meaning that X' can be determined when the dummy rate ρ can be estimated accurately.

We depict in Fig. 7 as a dark (vertical) short line the likely profiles X' that the OQF-PIR strategy might have transformed into the observed Y'. As we can see, the diversity of likely X' is rather limited, even when the estimation of ρ has low confidence (i.e., probability density of ρ with high variance).

The point marked as $\bullet$ in Fig. 7 corresponds to another possible observation $Y = (y_a, y_b, y_c)$ such that $y_a = y_b < y_c$. In this case, it is clear that the DGS is generating enough dummies to fill the weakest category (either a or b), but not enough to bring Y to Y^T. $W = (w_a, w_b, 0)$, with $w_a + w_b = 1$; and $\hat{x_c} = \frac{y_c}{1-\hat{\rho}}$. The space of likely real profiles $\hat{X}$ is depicted as a dark diagonal line in the upper right corner of Fig. 7. While this scenario leaves some room for uncertainty, we can see that the set of likely real profiles X is still rather limited.

Finally, we show in Fig. 8 a scenario in which the dummy rate ρ is sufficient for achieving $Y = Y^T$. We show as a dark inner triangle the space of likely profiles $\hat{X}$ that may have originated $Y = Y^T$ given $\hat{\rho}$. As we can see, even in this case OQF-PIR does not provide a high level of profile protection. Finally, we note that by using background information the adversary may be able to further reduce her uncertainty on X.

F. Noise Injection for Search Privacy Protection.

Lastly, we consider the Noise Injection for Search Privacy Protection (NISPP) strategy proposed by Ye et al. [30]. Similarly to Rebollo-Monedero and Forné [25], NISPP aims at finding the optimal dummy queries distribution amongst categories. The main difference with respect to [25] is that Ye et al. consider the mutual information between observed and real profiles $I(\mathcal{Y}; \mathcal{X})$ as optimization criteria. The optimal DGS is the one that brings $I(\mathcal{Y}; \mathcal{X})$ closer to zero, and when $I(\mathcal{Y}; \mathcal{X}) = 0$, the observed profile Y does not leak any information about the real profile X. With respect to the profile privacy properties defined in Sect. III-A, $I(\mathcal{Y}; \mathcal{X}) = 0$ corresponds to $\mathcal{E}_{\mathcal{Y}} = H(\mathcal{X})$, as $I(\mathcal{Y}; \mathcal{X}) = H(\mathcal{X}) - H(\mathcal{X}|\mathcal{Y})$, i.e., perfect privacy protection.

With respect to query privacy properties, NISPP assumes that dummy and real queries are indistinguishable based on their content and metadata (but provides no specifics on how this could be implemented in the DGS). Further, it considers that each possible query corresponds to a category of its own, with the goal of making their system robust to any SCA that could possibly be implemented by the adversary. Note that considering individual queries as categories implies that profile-based and query-based analysis are equivalent. Thus, $I(\mathcal{Y}; \mathcal{X}) = 0$ also corresponds to maximum deniability and unobservability of queries ($\mathcal{D} = \mathcal{U} = \Pr[D]$).

Ye et al. propose two DGS constructions, assuming that the user real profile X is available. The first DGS construction achieves $I(\mathcal{Y}; \mathcal{X}) = 0$ assuming that at least $N_Q - 1$ dummy queries are generated per real query (i.e., $\Pr[D] \geq \frac{N_Q - 1}{N_Q}$), where N_Q is the number of possible queries. For each real query the DGS is allowed to generate all other $N_Q - 1$ possible queries, and thus the strategy results in a uniform observed profile Y regardless of which

is the real profile X. This approach is however impractical in realistic settings where N_Q is large.

The second DGS construction proposed by Ye et al. considers that only one dummy query is generated per real query; i.e., $\Pr[D] = 0,5$, and proposes a (deterministic) algorithm that outputs the distribution of dummy queries that minimizes $I(\mathcal{Y};\mathcal{X})$, given X. The experimental results presented for $I(\mathcal{Y};\mathcal{X})$ however do not consider a strategic adversary who takes background information into account. A security evaluation of the proposed DGS would also involve (1) testing its robustness to SCAs that identify topics of interest and in turn feed this information to a DCA that distinguishes queries; and (2) studying whether the noise added by the DGS is predictable and invertible, such that a filtered profile Z can be constructed whose mutual information $I(\mathcal{Z};\mathcal{X})$ is larger than $I(\mathcal{Y};\mathcal{X})$ (or, in other words, such that $\mathcal{E}_{\mathcal{Z}} < \mathcal{E}_{\mathcal{Y}}$), further diminishing its privacy properties. Such comprehensive analysis of NISPP's second strategy is beyond the scope of this paper.

V. Summary of basic features in OB-PWS systems analysis

In the previous section we have described and analyzed a series of OB-PWS tools, and pointed out a variety of flaws in both their designs and evaluations that lead to an overestimation of the level of privacy that they offer. In this section we revisit our analysis and classify the reviewed OB-PWS systems according to their features, discussing the impact of each feature on the properties of the schemes.

Table II summarizes the main features considered in our evaluation. In this table ticks indicate that an OB-PWS system possesses a feature, and crosses that it does not. We write "**?**" when we have not evaluated the feature for a given system and we write "n/a" when the feature cannot be evaluated for a system due to a lack of specification in the original paper.

A. Dummy generation strategies

The dummy generation strategies DGS of the studied systems can be classified in two broad categories. On the one hand we have systems that focus on the obfuscation of the real profile as a whole, assuming that real and dummy queries are indistinguishable based on content and metadata. TMN [18], PRAW [28], OQF-PIR [25], and NISPP [30] fall into this category.

On the other hand we identify systems that focus on hindering the adversary's ability to distinguish real and dummy queries, assuming that query indistinguishability implies protection at a profile level. In this category we have TMN [18], GooPIR [9], PDS [23], and NISPP [30].

Note that we have classified TMN and NISPP in both categories. In TMN, the DGS is mostly focused on reducing the distinguishability of real queries, but dummies are selected in such a way that the profile observed by the adversary is different from the user's real profile. NISPP, as explained in Sect. IV-F, considers that each individual query corresponds to a category, and hence the query and profile properties are equivalent for this strategy.

B. Privacy Definitions

A second point in which the studied systems diverge is in the privacy property that they aim to achieve. Even though all schemes share a common objective, namely to prevent the adversary from learning the users' search interests, there are various ways in which they formalize this abstract privacy goal.

GooPIR and PDS are query-oriented schemes whose goal is to generate dummy queries that are hard to distinguish by the adversary, thus ensuring that user queries are k-deniable. In other words, these systems provide the user with an alibi with respect to which queries they have issued, and which queries have been issued by the OB-PWS tool.

GooPIR and PDS suggest that users can also claim that the profile recovered by the adversary does not reflect their interests, as it contains noise from dummy queries. However, it is unclear how this query k-deniability property relates to the amount of profile obfuscation provided by these systems —i.e., to what extent k-deniability prevents the adversary from inferring the topics of interest of a user.

Profile-oriented systems on the other hand tend to rely on privacy definitions that relate to the (dis)similarity of profiles. For TMN and PRAW privacy is related to the similarity between the real profile of the user and the profile available to the adversary. The more dissimilar these profiles are, the better the privacy protection provided by the system. OQF-PIR alternatively considers that privacy increases as the observed profile is more similar to the average population profile. Although PDS uses a query-based approach, its DGS takes into account semantic distance and generates dummies on topics that are as semantically distant as possible from the topic of the real query —thus reducing the similarity between the real and observed profiles.

These approaches implicitly assume that there is a direct correlation between the privacy offered by the system and the similarity between the observed and the real (or the observed and the population) profiles. Nevertheless, we have shown (see Sect. IV-D and Sect. IV-E) that distance-based metrics do not necessarily reflect the privacy protection provided to profiles, as they are not indicative of how much the adversary knows about the real user profile.

Finally, NISPP uses mutual information as privacy metric, and its DGS aims at obfuscating the real profile such that the observed profile leaks no information about it. We recall that this metric is equivalent to the equivocation $\mathcal{E}_{\mathcal{Z}}$ (introduced in Sect. III-A), which measures the uncertainty of the adversary on real profiles X given the filtered profile. The average amount of profile information leaked by the DGS can be computed as $H(\mathcal{X}) - \mathcal{E}_{\mathcal{Z}}$.

Table II
OB-PWS TOOLS: SUMMARY OF FEATURES.

		TMN [18]	GooPIR [9]	PDS [23]	PRAW [28]	OQF-PIR [25]	NISPP [30]
DGS	Profile oriented	✓	✗	✗	✓	✓	✓
	Query oriented	✓	✓	✓	✗	✗	✓
Privacy definitions	Privacy as (dis)similarity	✓	✗	✓	✓	✓	✗
	Privacy as query k-deniability	✗	✓	✓	✗	✗	✗
	Privacy as information leakage	✗	✗	✗	✗	✗	✓
Analysis	Aware adversary	✗	✓	✓	✓	✗	✓
	Considers background information	✗	✓	✗	✗	✓	✗
	Considered strategic adversary	✗	✓	✗	✗	✗	✗
	Exploitable query content	✓	✓	?	?	n/a	n/a
	Exploitable query metadata	✓	✗	✗	n/a	n/a	n/a
	Invertible DGS profile transformation	?	?	?	✓	✓	?

C. Analysis and evaluation

Systems also differ in their assumptions on the capabilities and knowledge of the adversary. TMN and OQF-PIR consider that the adversary is *not* aware of users having installed an OB-PWS tool. This is reflected in the security evaluation that accompanies the description of the designs, which is non-existent in TMN and flawed in OQF-PIR, as we have shown in Sect. IV-E.

The reviewed systems vary widely in their assumptions on background knowledge. OQF-PIR assumes that the population profile is available to both the DGS and the adversary. GooPIR assumes that the frequency of appearance of search keywords in the Web is available to the tool, and also used by the adversary to attempt to distinguish between real and dummy queries. TMN, PDS, and PRAW neglect in their evaluation the fact that the adversary may have access to background information on likely user profiles —even although it has great impact on the security they offer (as illustrated in Sect. IV-D). Lastly, NISPP's analysis (explicitly) does not take adversarial background information into account (though acknowledging that background information would diminish the level of privacy protection offered), while considering that the profile of the user is available to the DGS.

Of all the studied schemes, only GooPIR's evaluation considers a strategic adversary that tries to attack the implemented DGS. Neglecting the adversary's knowledge of the dummy generation strategy results in an overestimation of the privacy provided by the system. We demonstrate the negative effects of such disregard on our analysis of PRAW and OQF-PIR (Sect. IV-D and IV-E, respectively) where we show how the adversary can invert the obfuscation algorithm and gain information about the real profile.

Dummy query filtering is possible in TMN given the keyword popularity, semantics [2], or grammatical construction [5] of dummy queries. GooPIR protects individual queries against attacks that exploit the popularity of the keywords in the Web, but it is vulnerable to attacks that consider sequences of queries and exploit their semantic relationships. PDS attempts to prevent these attacks by canonizing queries, and generating sequences of dummy queries that are semantically related. The security of this strategy however relies heavily on a semantic classification algorithm SCA_{PDS}, and does not necessarily guarantee that a different SCA (with a different definition of "topics") will not distinguish dummy queries based on semantic correlations. PRAW aims at preventing query content attacks by selecting the keywords for its dummy queries on the "general" topics of interest for the user (but on different "specific" topics). PRAW's strategy for generating queries is however not sufficiently specified to allow for a thorough evaluation. OQF-PIR and NISPP are not concerned with individual queries and do not provide any specifics on how to generate dummy query content.

TMN specifies several strategies for generating dummy query metadata (headers, cookies). These strategies can however be exploited by an adversary to distinguish dummy and real queries. GooPIR and PDS send queries in batches of k (one real and $k-1$ dummy) such that query timing or metadata cannot be exploited for distinguishing queries. PRAW, OQF-PIR, and NISPP do not specify any strategies for generating query metadata.

PRAW and OQF-PIR present strategies to obfuscate the user profile using a specific profile transformation function: maximizing cosine similarity with the observed profile, and making the observed profile as similar as possible to the average population profile, respectively. We show how these strategies allow the adversary to predict and (partially) reverse the transformation. NISPP's first (impractical) construction consists in making the profile appear as uniform by generating $N_Q - 1$ dummy queries for each query issued by the user, where N_Q is the number of possible queries. The second (practical) construction would require additional analysis, as mentioned in Sect. IV-F. Similarly, analyzing the effectiveness of profile filtering algorithms for TMN, GooPIR, and PDS, would require studying how these tools introduce noise in the observed profiles under different SCAs. If the distortion introduced is predictable (i.e., if there is a consistent pattern in how noise is added to profiles), the

adversary may be able to implement PFAs that filter out (part of) the noise introduced by the dummy queries in the observed profile.

Finally, we would like to highlight that none of the security evaluations presented with the reviewed systems was done from both a query-based and a profile-based perspectives —thus overlooking potential vulnerabilities. As we have pointed out in our analysis framework, performing both a query-based and a profile-based analysis is crucial for a comprehensive evaluation of the privacy properties offered by a OB-PWS design.

VI. Challenges and Open Problems

We have stated that an effective DGS should ensure that real and dummy queries are indistinguishable. Several of the studied systems [9], [18], [28] propose to use a predefined lexicon. We have shown that this feature can be exploited by a DCA to distinguish real queries formed by keywords that are not part of the lexicon. An approach that constructs the lexicon in a way that it is difficult for the the adversary to predict which keywords are included in it could mitigate this problem. Another possible countermeasure is to map query keywords to the words in the predefined lexicon, as the canonical queries proposed in [23]. This strategy indeed counters the aforementioned attack, but its viability in a practical scenario is dubious. Canonical queries reduce the utility of the search results as they cannot be as specific as the original queries. This effect is even more serious when queries refer to keywords difficult to canonize, e.g., proper nouns.

The evaluation of a DGS should consider the prior probability of a given query and also its posterior probability given the sequence of preceding queries. The DGS should mimic users' behavior in terms of query timing, metadata, semantics, and grammar, amongst other exploitable features [2], [5], [24]. Furthermore, related visible actions such as links that have been clicked after the search results have been returned to the user should also be taken into account. Designing a DGS that outputs plausible dummies indistinguishable from real queries and mimics other relevant aspects of user behavior is far from trivial and still one of the main challenges of OB-PWS.

Several of the analyzed systems [23], [25], [28] base their dummy generation strategy on a given SCA_{DGS}, and evaluate the privacy protection they offer assuming that the adversary uses the same semantic classification algorithm. This does not consider attacks in which the adversary uses a semantic classification different from SCA_{DGS} for recovering the profile. The design of DGS strategies that are safe against such attacks is a hard problem, as it is very difficult to predict what SCA the adversary will use. We note that this problem was already acknowledged in [30] by Ye et al. who alert of the negative consequences that the attack could have on the privacy protection provided by their tool.

In this paper we have considered that the output of a DCA is a binary classification of queries; either as real or dummies. An alternative approach would be to consider a probabilistic DCA that assigns to each query probability of being real (or dummy). These probabilities can then be used to assign weights to categories when reconstructing the user profile.

We have analyzed systems from a query-based and a profile-based perspectives. We have found that query-based privacy, usually formalized as query k-deniability, is well understood. On the other hand we have found that profile-based properties seem to be much harder to articulate. We have indicated that distance-based metrics fail to capture privacy notions, and that designing the DGS to maximize (or minimize) a distance metric is a fundamentally flawed approach, as it enables the adversary to predict (and remove) the noise introduced in the observed profile.

We have proposed to use information theoretic metrics (similar to those introduced by Ye et al. [30]) to model the information leaked by the different dummy generation strategies. Nevertheless, we acknowledge that the use of such metrics on deployed systems entails some challenges. First, the probability distribution associated to the random variable $\mathcal{X}$ may not be available to the system designer, who may only have access to an approximation (e.g., profiles constructed from observed queries over a limited period of time). A more suitable metric should consider the effect of considering this approximation on the measured privacy level. Secondly, as mentioned in Sect. III-A, the conditional entropy is an average measurement of the privacy protection provided by an OB-PWS tool. This should be taken into account when evaluating the system, so as to guarantee a minimum level of privacy protection to all users. Complementary metrics should be considered to provide a measure of the *worst-case* profile protection provided by a DGS, for instance the conditional min-entropy:

$$H_\infty(\mathcal{X}|\mathcal{Z}) = -\log(\max_{X\in\mathcal{X},Z\in\mathcal{Z}}\{\Pr[\mathcal{X}=X|\mathcal{Z}=Z]\})\,.$$

Perfect privacy protection from an information-theoretic perspective may be impractical to achieve in reality. Further, it is unclear that complete concealment of the profile is a requirement for all users and applications. Therefore it may be desirable to define metrics that measure information leakage with respect to less demanding privacy requirements, such as altering the observed level of interest in specific categories. An interesting approach would be to let users indicate the type of profile they would like to present to the search engine and generate the dummy queries accordingly. Profile privacy metrics in this case should express the extent to which the adversary is able to detect and reverse the noise introduced in the profile categories whose weight has been modified.

We have highlighted the importance of carrying out both profile-based and query-based analyses when evaluating a

DGS. Nevertheless it should be taken into account that depending on the application the privacy goal of the system may be more focused on profile-based or query based properties. A system may for instance focus on preventing the disclosure of the search interests of the user but not be necessarily concerned about specific queries. Conversely, the goal of the system may be to prevent the adversary from learning whether or not specific queries are real but not necessarily concerned about the general interests of the user. As an example, an HIV-positive user may be interested in concealing that her HIV-related queries are real, or that she is interested in health-related topics in general. The former refers to concealing specific queries, thus requires a query-based approach; whereas the latter refers to concealing general interests thus it seems more appropriate to choose a profile-based approach. Regardless of the approach chosen in the design of the system we must stress that the analysis of the scheme must take into account strategic adversaries that know the dummy generation strategy and try to defeat it from *both* a profile and a query perspective, as vulnerabilities detected by an profile-based analysis may influence the query-based privacy properties, and vice versa.

Our analyses reveal that a strategic adversary can exploit certain types of dependencies of the dummy generation strategy on the user profile or on real queries. Nevertheless, our results do not allow us to extract conclusions about which types of dependencies result in the better or worse privacy protection. The optimal design decisions with respect to such dependencies in order to obtain an effective and robust OB-PWS tool remains as an open question.

Some of the systems we have studied implicitly assume that the adversary is unaware of the use of the OB-PWS tool [18], [25]. In other words, they assume that the tool is *unobservable* for the adversary and hence she shall not try to invert the effect of the dummy generation strategy. While such a property may be desirable we argue that achieving unobservability is non-trivial and cannot be taken as granted without a proper analysis. Techniques to construct and analyze unobservable OB-PWS tools are left as an open problem.

A related problem is whether the dummy queries should contain controversial keywords, e.g., “bomb”, “HIV”, or “gay marriage”. If the tool is unobservable and such keywords are included, users may appear as involved in subversive activities, having a particular disease, or having certain sexual orientation, which may be undesirable in certain situations. The opposite strategy (avoiding such keywords in dummy queries) puts users in a delicate position: either they expose themselves; or they refrain from issuing queries related to sensitive topics, effectively acting as a censors on their own queries [17]. We note that this self-censorship conflicts directly with the purpose of private web search, that is to allow users to freely search for information without revealing their preferences.

The above problems are alleviated when the tool is observable and dummy queries can contain controversial keywords. In this case the user can plausibly claim that queries containing these keywords were originated by the OB-PWS tool On the other hand, if sensitive terms are not included in the OB-PWS lexicon the user is again subject to self-censorship, reducing the utility of the system. Finding the optimal balance between these properties is extremely challenging as the decision not only depends on technical possibilities but also on subjective opinions particular to each individual.

VII. Conclusion

In this paper we have reviewed the state of the art in obfuscation-based private web search (OB-PWS) techniques. Our study contributes towards systematizing existing knowledge by improving the understanding of the conceptual building blocks of OB-PWS systems; defining and formalizing relevant privacy properties; and outlining the elements that must be taken into account in their security evaluation.

We have proposed an abstract model that captures the key elements and processes in OB-PWS systems, and an analysis framework that considers privacy properties associated to both search profiles and individual queries. Using this framework we have analyzed six proposed OB-PWS strategies and found vulnerabilities that had not been taken into account in their original security evaluations —implying that the level of privacy offered by these systems was being overestimated.

Further, we have identified a series of features that should be considered in a systematic security evaluation of OB-PWS systems. In particular, we argue that OB-PWS proposals should be analyzed with respect to both profile-based and query-based privacy properties regardless of the design principles and privacy goals of the scheme. It is our hope that our results will serve as guidance for the designers of future robust and effective OB-PWS tools.

Acknowledgments

This work was supported in part by the projects: GOA TENSE (GOA/11/007), IAP Programme P6/26 BCRYPT, EC ICT-2007-216676 ECRYPT II, IWT SBO SPION, FWO G.0360.11N, and FWO G.068611N. C. Troncoso and C. Diaz are funded by the Fund for Scientific Research in Flanders (FWO).

References

[1] Dakshi Agrawal and Dogan Kesdogan. Measuring anonymity: The disclosure attack. *IEEE Security & privacy*, 1(6):27–34, 2003.

[2] Rami Al-Rfou’, William Jannen, and Nikhil Patwardhan. TrackMeNot-so-good-after-all. Technical report, Stony Brook University, December 2010.

[3] Oliver Berthold, Hannes Federrath, and Stefan Köpsell. Web mixes: A system for anonymous and unobservable internet access. In Hannes Federrath, editor, *Design Issues in Anonymity and Unobservability*, volume 2009 of *LNCS*, pages 115–129. Springer, 2000.

[4] Jordi Castellà-Roca, Alexandre Viejo, and Jordi Herrera-Joancomartí. Preserving user's privacy in web search engines. *Computer Communications*, 32(13-14):1541–1551, 2009.

[5] Richard Chow and Philippe Golle. Faking contextual data for fun, profit, and privacy. In Ehab Al-Shaer and Stefano Paraboschi, editors, *ACM Workshop on Privacy in the Electronic Society (WPES 2009)*, pages 105–108. ACM, 2009.

[6] Thomas M. Cover and Joy A. Thomas. *Elements of information theory*. Wiley-Interscience, 1991.

[7] George Danezis. Statistical disclosure attacks. In Dimitris Gritzalis, Sabrina De Capitani di Vimercati, Pierangela Samarati, and Sokratis K. Katsikas, editors, *SEC*, volume 250 of *IFIP Conference Proceedings*, pages 421–426. Kluwer, 2003.

[8] Roger Dingledine, Nick Mathewson, and Paul F. Syverson. Tor: The second-generation onion router. In *13th USENIX Security Symposium*, pages 303–320. USENIX, 2004.

[9] Josep Domingo-Ferrer, Agusti Solanas, and Jordi Castellà-Roca. *h(k)*-private information retrieval from privacy-uncooperative queryable databases. *Online Information Review*, 33(4):720–744, 2009.

[10] Peter Eckersley. How unique is your web browser? In *Proceedings of the 10th international conference on Privacy enhancing technologies*, PETS'10, pages 1–18, Berlin, Heidelberg, 2010. Springer-Verlag.

[11] Yuval Elovici, Chanan Glezer, and Bracha Shapira. Enhancing customer privacy while searching for products and services on the world wide web. *Internet Research*, 15(4):378–399, 2005.

[12] Yuval Elovici, Bracha Shapira, and Adlai Maschiach. A new privacy model for hiding group interests while accessing the web. In Sushil Jajodia and Pierangela Samarati, editors, *WPES*, pages 63–70. ACM, 2002.

[13] Yuval Elovici, Bracha Shapira, and Adlai Maschiach. A new privacy model for web surfing. In Alon Y. Halevy and Avigdor Gal, editors, *NGITS*, volume 2382 of *Lecture Notes in Computer Science*, pages 45–57. Springer, 2002.

[14] Yuval Elovici, Bracha Shapira, and Adlay Meshiach. Cluster-analysis attack against a private web solution (PRAW). *Online Information Review*, 30(6):624–643, 2006.

[15] Robert G. Gallager. *Information Theory and Reliable Communication*. John Wiley & Sons, Inc., New York, NY, USA, 1968.

[16] Ian Goldberg. Improving the robustness of private information retrieval. In *IEEE Symposium on Security and Privacy (S&P 2007)*, pages 131–148. IEEE Computer Society, 2007.

[17] Seda Gürses. Privatsphäre und praktiken digitaler kontrolle. *Demokratie... in der neuen Gesellschaft, Informationen aus der Tiefe des Umstrittenen Raums*, 2007.

[18] Daniel C. Howe and Helen Nissenbaum. TrackMeNot: Resisting surveillance in web search. In Ian Kerr, Valerie Steeves, and Carole Lucock, editors, *Lessons from the Identity Trail: Anonymity, Privacy, and Identity in a Networked Society*, chapter 23, pages 417–436. Oxford University Press, Oxford, UK, 2009.

[19] Rosie Jones, Ravi Kumar, Bo Pang, and Andrew Tomkins. "I know what you did last summer": query logs and user privacy. In Mário J. Silva, Alberto H. F. Laender, Ricardo A. Baeza-Yates, Deborah L. McGuinness, Bjørn Olstad, Øystein Haug Olsen, and André O. Falcão, editors, *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management (CIKM 2007)*, pages 909–914. ACM, 2007.

[20] Tsvi Kuflik, Bracha Shapira, Yuval Elovici, and Adlai Maschiach. Privacy preservation improvement by learning optimal profile generation rate. In Peter Brusilovsky, Albert T. Corbett, and Fiorella de Rosis, editors, *User Modeling*, volume 2702 of *Lecture Notes in Computer Science*, pages 168–177. Springer, 2003.

[21] Eyal Kushilevitz and Rafail Ostrovsky. Replication is not needed: single database, computationally-private information retrieval. In *IEEE Annual Symposium on Foundations of Computer Science (FOCS 97)*, pages 364–373, 1997.

[22] Mummoorthy Murugesan and Chris Clifton. Providing privacy through plausibly deniable search. In *SDM*, pages 768–779. SIAM, 2009.

[23] Mummoorthy Murugesan and Christopher W. Clifton. Plausibly Deniable Search. In *Proceedings of the Workshop on Secure Knowledge Management (SKM 2008)*, November 2008.

[24] Sai Teja Peddinti and Nitesh Saxena. On the privacy of web search based on query obfuscation: A case study of TrackMeNot. In Mikhail J. Atallah and Nicholas J. Hopper, editors, *Privacy Enhancing Technologies*, volume 6205 of *LNCS*, pages 19–37. Springer, 2010.

[25] David Rebollo-Monedero and Jordi Forné. Optimized query forgery for private information retrieval. *IEEE Transactions on Information Theory*, 56(9):4631–4642, 2010.

[26] Michael K. Reiter and Aviel D. Rubin. Anonymous web transactions with crowds. *Commun. ACM*, 42(2):32–38, 1999.

[27] Claude Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423:623–656, 1948.

[28] Bracha Shapira, Yuval Elovici, Adlay Meshiach, and Tsvi Kuflik. PRAW - A PRivAcy model for the Web. *JASIST*, 56(2):159–172, 2005.

[29] Bill Tancer. *Click: What Millions of People Are Doing Online and Why it Matters*. Hyperion, 2008.

[30] Shaozhi Ye, Shyhtsun Felix Wu, Raju Pandey, and Hao Chen. Noise injection for search privacy protection. In *CSE (3)*, pages 1–8. IEEE Computer Society, 2009.