Towards a theory of variable privacy

Poorvi L. Vora
Hewlett-Packard Co., USA.
Phone: 541-715-6509
Fax: 541-715-4183
poorvi@acm.org
poorvi@ieee.org

This is a modified version of a journal paper in review. This draft last revised 7 May 2003.

Initial versions of this material may also be found in
- Poorvi Vora, "The channel coding theorem and the security of binary randomization", to be presented at *Int. Symposium on Info. Theory, 2003*, and
-Poorvi Vora, "A non-binary approach to privacy using randomization", expanded abstract in review.

Keywords: privacy, randomization, noisy channel, channel codes, reliable attacks, non-perfect protocols, information-theoretic security, theory of statistical attacks, channel coding theorem, source-channel separation

Abstract

We define "variable privacy" as the use of non-perfect protocols with parameters controlled by Alice. Variable privacy enables Alice to choose the amount of information leaked to Bob, in situations where information revelation bears a privacy cost and also provides a benefit. We propose a framework for the study of variable privacy, using a security perspective to obtain a privacy measure of the binary symmetric randomization protocol (flipping a bit with probability $1-\rho$). We define an attack as any sequence of protocol instances that decreases estimation error on a bit beyond that possible with a single instance. Viewing the protocol as a communication channel for the data to be protected, we show that channel codes - i.e. error-correcting and error-detecting codes - are particularly efficient attacks. In particular, they can be more efficient than the repeated query attack.

The cost of a repeated query attack, per bit of entropy in the data, increases monotonically with a decrease in error. We show that attacks other than the repeated query attack can achieve zero asymptotic error with merely a constant cost per bit of entropy; that the cost is asymptotically tightly bound below if asymptotic error is zero; that polynomial time attacks that achieve the bound exist; and that similar attacks can be constructed in linear time. We define the privacy measure of randomization as the tight asymptotic lower bound on the number of protocol instances required for zero asymptotic error, per bit of entropy. We show that the privacy measure of a randomization protocol is the inverse of its capacity if it is viewed as a communication channel. Our results follow easily from the channel and source-channel coding theorems of Shannon, but we are not aware of any other work that uses these or similar results to study the security of any cryptographic protocols. Our results, in addition to providing a measure of the privacy cost of the protocol to Alice, have implications for the security of statistical databases.

1 Introduction

From the point of view of the traditional theory of security, Alice either trusts another entity, Bob, or does not. Accordingly, she either completely reveals a piece of information to him or not. When she does reveal information, she uses protocols that allow no information leakage to untrusted parties; for example, she might encrypt information so only Bob can decrypt it. A broader approach is required to consider information revelation when it provides both benefit and privacy invasion, and when parties are not easily classified as trusted or untrusted. As an example of a situation where information revelation provides benefit yet also results in privacy invasion, consider recommender systems [20], which can provide Alice with personalized recommendations in return for ratings; personalized search results in return for usage history; or personalized storefronts in return for history with the store. Making sense of large amounts of information can be quite difficult without personalization, hence choosing not to reveal information might correspond to loss of benefit to Alice; on the other hand, revealing her entire usage profile would result in a loss of privacy. In this scenario, allowing Alice the binary choice of revealing, or not revealing, specific information is not sufficient; she needs to be able to reveal "some information" for "some benefit".

Randomization of personal information, with parameters controlled by the data collector, has recently been proposed as a means of personal privacy protection in data mining applications [2, 3] which require the statistics of the data and not accurate individual values. The larger the probabilistic perturbation of the data, the more privacy provided to individual points, and the less accurate the statistics. This technique, also known as "data perturbation", has been in use for about twenty years in statistical database security [1]. The problem it addresses in statistical databases is described in the next paragraph.

The purpose of a statistical database is to provide statistics to researchers while keeping individual values "private". For example, a health database would keep "private" whether individual X had Hepatitis A or not, but would reveal how many members in a community had Hepatitis A. One way of stating the technical problem is as follows: $f(x_1, x_2, ...x_k)$ is to be revealed without revealing anything more than can be determined by knowledge of $f(x_1, x_2, ...x_k)$. This technical problem is easily solved using zero-knowledge secure computation protocols. However, this technical statement of the problem has been determined to be insufficient by the database community. It is well-known that "trackers", i.e. dishonest data collectors, can make many queries of the database to determine the values of x_i. In other words, trackers can request $A_i = f_i(x_1, x_2, ...x_k)$, $i = 1, 2, ...n$. These provide simultaneous equations in $x_1, x_2, ...x_k$. The equations are consistent and can be solved simultaneously to obtain $x_1, x_2, ...x_k$, without any information other than $A_i, i = 1, 2, ..n$. The randomization of the x_i or the A_i before revealing the values to the data collector would make the trackers task more difficult by making the equations considerably more inconsistent.

Another way of looking at the problem can be illustrated through looking at binary A_i. In this case, revealing the values accurately corresponds to revealing at most one bit of information with each protocol instance. Revealing the values inaccurately is, we will show, equivalent to revealing at most a *fraction of a bit* of information. The value of the fraction, we will show, depends on the noise. Hence, a better technical model for the statistical database security problem is: $f(x_1, x_2, ...x_k)$ is to be revealed to a probabilistic uncertainty pre-determined and known to the data collector. This technical model is the one used in statistical databases for about two decades. Zero-knowledge protocols may be used to compute the perturbed values of A_i. The perturbation, if the A_i are binary, cannot correspond to a probability of error of 0.5, as this would reveal no information about A_i.

It is clear that randomization leaks information, and, viewed as a primitive by itself, it does not provide perfect secrecy. It is not intended to. The amount of information it leaks is related to the probabilistic perturbations. It provides a good model to approach the question described by us at the beginning of this paper - that of providing Alice the choice of revealing "some" information for "some" benefit when parties are not easily classified as trusted or untrusted. Alice obtains a certain level of privacy by choosing a certain value of the randomization parameters (in all current applications of randomization, including those in databases and data mining, Alice does not control the information leakage). We define "variable privacy" as the use of non-perfect protocols, such as randomization, with user-controlled parameters, for the protection of personal information. It would be used to trade off privacy for benefits, such as more or less accurate recommendations based on more or less accurate usage profiles.

Individuals are used to trading privacy for benefit in the physical world - for example, in a number of western countries, individuals are willing to participate in grocery store discount card programs, trading their grocery profiles for a discount. At the same time, they are also willing to pay a few dollars a month to be kept off phone directories; paying for their privacy and forgoing the benefit of being listed. In this market, randomization provides a range of choices for Alice. For example, she might reveal a randomized grocery profile for a lower discount. Randomization can also be used to provide a negotiation protocol [14]. For example, Bob approaches an airline, offering a certain amount of money for a seat on a particular flight on a particular date. The airline responds "Certainly not. We have only $x \pm \frac{d}{2}$ tickets left on that flight". This response could be certified by, for example, a trusted computing platform. Last, randomization makes it more difficult for colluding data collectors to piece together the profile of an individual from values gathered by different entities at different times.

The randomization protocol bears some similarity to the universal oblivious transfer of Brassard and Crepeau [6], and the generalized oblivious transfer of Brassard et al [7] - it is the probabilistic perturbation of a function of stored data points. The transfer, however, is not oblivious, i.e. Alice knows what was sent. The randomization protocol we consider in this paper, the binary symmetric randomization protocol (flipping a bit with probability $1-\rho$), is similar to the "ρ-noisy transfer" (or the α-slightly oblivious transfer) of Crepeau and Kilian [10], except, again, the protocol we consider is not oblivious. Further, in our case, the use of a sequence of protocol instances is set up in an adversarial model, and the randomization can change from a single execution of the protocol to the next.

A satisfactory security analysis of the randomization primitive does not exist. We provide a security analysis of Our analysis provides the cost of an attack to Dishonest Bob, and the corresponding privacy measure for Alice. (Attacks are known to exist because the randomization protocol, in general, is computationally and information-theoretically non-perfect). We consider a most-powerful Bob and a most-naive Alice, consistent with wanting to provide security bounds on the protocol. Most of our results are easily extended to continuous and discrete protocols.

1.1 The protocol

The protocol we consider is as follows:

1. Bob asks Alice the value of a variable, $X \in \mathcal{X} = \Sigma = \{0, 1\}$.

2. Bob receives the variable $Y \in \mathcal{Y} = \Sigma$ generated according to $P(Y|X)$,

$$P(Y|X) = \begin{cases} \rho & \text{Y=X} \\ \text{1-}\rho & \text{Y} \neq X \end{cases}$$

3. There are no conditions on the amount of information Alice has about what Bob received.

We refer to X as the *requested bit*, and Bob's request as a *query* of X. Y is the *randomized response* to Bob's request, denoted $\phi(X)$, where ϕ denotes the randomization. Each protocol instance corresponds to the completion of a single query, and *query complexity* refers to the number of protocol instances. We define the collection of bits Bob is interested in as a profile, $P \in \mathcal{P}$.

ρ is the *probability of truth*, and $1-\rho$ the *probability of a lie.* As long as $P(Y|X) \neq P(Y)(\rho \neq 0.5)$, the uncertainty in X is reduced on knowledge of Y, and the protocol leaks some information about X.

In traditional cryptographic terms, X is the plaintext; the protocol a one-time pad; Y the ciphertext; and the key bits are independent, identically distributed, with $P(0) = \rho$, $P(1) = 1-\rho$. The length of the key is the length of the plaintext, but, when $\rho \neq 0.5$, the entropy in the key is strictly smaller than its length, and protocol security is not perfect. Hence, in each instance of the protocol, there is "room" in the single bit of ciphertext to transmit a small amount of information - the difference between unity and the ratio of key entropy to key length. In this paper, we present results on exactly how Dishonest Bob can optimally use this "room" to transmit information about the value of the plaintext. We also present a tight asymptotic upper bound on the efficiency with which he can transmit information; in particular, we show that, on average, he can use all of the "room" each time the protocol is executed. This upper bound is similar to the unicity distance [24], which is calculated assuming there is "room" in correlated plaintext to carry information about the key. We describe how our bound can be achieved, using a technique quite different from, and more efficient than, the one described in [24].

We propose that Alice, armed with our results, be allowed to participate in the choice of $P(Y|X)$ and thus negotiate an appropriate cost-benefit trade-off for herself in situations where she trades privacy for benefit.

1.2 Our approach

Our approach is based on the first of Shannon's papers on a mathematical theory of communication [23], and connects his work on communication in the presence of noise to his work on secrecy. We are not aware of any other work that does so. A perfectly secret protocol, by definition, is one in which the *a posteriori* and *a priori* distributions of the data are identical (information-theoretic secrecy) or indistinguishable (computational secrecy). A communication channel has zero capacity if and only if the *a posteriori* and *a priori* distributions of the data are identical (see the appendix for a definition of capacity). Thus a non-perfect protocol, by definition, has non-zero capacity if viewed as a communication channel for the information to be protected. The binary symmetric randomization protocol is a communication channel for the plaintext, and Bob, particularly in his dishonest incarnation, is interested in transmitting the plaintext efficiently. Channel codes, because they are among the most efficient means of communicating over channels, correspond to efficient attacks; in particular we show that they correspond to adaptive and non-adaptive related plaintext attacks.

Bob does not have access to the bits before they are randomized. He can, however, encode the messages he wishes to have transmitted over the channel, because the protocol allows him to choose a pattern among his queries, and hence, among plaintext bits. This corresponds to allowing him to choose a code (which is essentially a pattern among the bits of a codeword). The coded messages correspond to sequences of related queries, or related plaintext. Thus, through controlling the queries, Bob ensures that Alice encodes plaintext before sending it over the channel. The channel coding and source-channel coding theorems provide tight bounds on the efficiency of communication over a given channel. Transmission of information over the protocol-channel (including attacks) is governed by these theorems too, and we use them to obtain tight upper bounds on the efficiency of attacks.

We define an attack as any sequence of queries that reduces estimation error beyond that associated with a single protocol instance; i.e. an attack corresponds to a choice of plaintext that reduces error on a single bit of the message beyond that implied by the probability of a lie. Bob can reduce error by repeatedly asking for the same bit (a repeated plaintext attack). Not only that, he can decrease error without bound by increasing queries without bound, i.e. if ω_n is the probability of error using n repeated queries, $n \to \infty$ implies $\omega_n \to 0$. Further, this is the best he can do with repeated queries, i.e. $\omega_n \to 0$ implies $n \to \infty$, and the cost per independent message bit, n, increases indefinitely if error is not bounded below.

We show that Bob can reduce error arbitrarily with *fixed, finite* query complexity per independent bit if he and Alice are willing to participate in a large enough number of queries. More specifically, for k independent bits (plaintext with entropy k), we demonstrate the existence of attacks in which $\omega_n \to 0$ as $n \to \infty$ for $\frac{n}{k}$ finite. This implies that the cost to Bob of decreasing error indefinitely, per independent message bit, $\frac{n}{k}$, is *bounded above*. Further, we demonstrate that if asymptotic error is desired to be zero, under reasonable assumptions, $\frac{n}{k}$ is asymptotically tightly bounded below; i.e. there is an asymptotically minimum achievable cost per independent bit if $n \to \infty \Rightarrow \omega_n \to 0$. Note that the lower bound obviously does not hold for attacks that do not seek to reduce error arbitrarily; a sequence of two repeated queries is an attack, but two queries may not be sufficient to reduce error arbitrarily. We define this minimum cost per independent bit as the privacy measure of the protocol, and show that it is the inverse of the channel capacity of the protocol viewed as a communication channel. We obtain our results by using Shannon's channel coding and source-channel coding theorems [23], and Fano's inequality [9, pg. 205].

The unicity distance, the expected number of plaintext bits required to determine a key when the redundancy of plaintext is known, is defined by Shannon [24] as the ratio of key entropy to language redundancy. Our bound - the number of plaintext bits required per message for zero asymptotic error, which we show is the entropy of the message divided by the channel capacity of the randomization protocol, is very similar to the unicity bound when message and key are switched. In the classical problem of decrypting when there is redundancy in the language, the bits of the key are completely random, i.e. its entropy is equal to its length. The message bits, however, are related and the entropy of the message is strictly smaller than its length, the difference per bit of message length is defined as the redundancy. Information on the key is carried in the ciphertext, and the most that can be carried per encrypted bit, on average, is equal to the redundancy in the plaintext. In our problem, the entropy of the key is strictly smaller than its length, and information about plaintext is carried in the ciphertext because the reduced entropy of the key allows for that.

Further, our model allows for attacks that use different messages, whereas the work in [24] uses the same key.

Our results follow very easily from Shannon's channel coding theorems. Our view of the protocol as a channel, however, has one important point of difference from the view of a channel in communication theory. The goal of communication theory is to increase information transfer over a channel given certain constraints. The goal of a privacy protocol is to decrease the information transfer over the protocol given certain constraints (such as the error in recommendations that use these perturbed data points). Because of this, Alice would be interested in channels with small capacity, i.e. "good" privacy protocols. On the other hand, Dishonest Bob is interested in the efficient transfer of plaintext over a particular protocol, typically a channel with small capacity, and a number of the constructive results from the theory of coding are of interest to him. Hence forth, we do not use the terms "queries" or "requested bits", but use "plaintext" instead.

1.3 Our main results

For definitions of the terms used below, please refer to section 3.

Theorem 1 *Given a set of equal-length messages $\mathcal{M}$, and a randomization protocol Φ, a one-to-one correspondence exists between (a) the set of all $(|\mathcal{M}|, n)$ binary channel codes (with feedback) on set of messages $\mathcal{M}$, for channel Φ, and (b) the set of all $(|\mathcal{M}|, n)$ deterministically-related plaintext attacks (deterministically-related adaptive plaintext attacks) on Φ using $\mathcal{M}$. The correspondence preserves rate and error.*

Theorem 2 *For a binary symmetric randomization protocol Φ, $\forall R < \mathcal{C}(\Phi)$, given a sequence of sets of equal-length messages, $\{\mathcal{M}_n\}$, $|\mathcal{M}_n| = 2^{Rn}$, there exists a reliable deterministically-related plaintext attack of rate R on Φ using $\{\mathcal{M}_n\}$.*

Theorem 3 *The rate of a reliable (M, n) probabilistically-related plaintext attack on Φ is bounded above by $\mathcal{C}(\Phi)$. Further, the rate of any small error probabilistically-related plaintext attack is asymptotically bounded above by $\mathcal{C}(\Phi)$.*

Theorem 4 *The tight asymptotic lower bound on the plaintext length, on average, per message, for a small error probabilistically-related plaintext attack, is $\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ if the message sequence is stationary, i.e.*

$$Lim_{n\to\infty}\omega_n \to 0 \Rightarrow Lim_{numberofmessages\to\infty}\frac{plaintextlength}{numberofmessages} \geq \frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}.$$

Corollary 1 *The tight asymptotic lower bound on the plaintext length, on average, per message, for a small error probabilistically-related plaintext attack on the binary symmetric randomization protocol with small bias β, $\Phi_{\mathcal{B}}(0.5 \pm \beta)$, is $\frac{ln2\times\mathcal{H}(P)}{2\beta^2}$ if the message sequence is stationary.*

Corollary 2 *The tight upper bound on the plaintext length, on average, per message, for any probability of error using protocol Φ, is $\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ if the message sequence is stationary.*

Corollary 3 *The privacy of Φ is $\frac{1}{\mathcal{C}(\Phi)}$.*

Corollary 4 *The privacy of $\Phi_{\mathcal{B}}(0.5 \pm \beta)$ is $O(\frac{1}{\beta^2})$.*

1.4 Organization of paper

In Section 2 we present example typical attacks, which motivate most of the definitions of Section 3. Section 4 presents our results with proofs. The conclusions are presented in Section 5, and the Appendix contains common definitions and proof sketches of important results from information theory.

2 Some example attacks

In this section we provide examples of three types of attacks on the randomization primitive of section 1.1. The first type is the repeated plaintext attack. We examine the properties of this type of attack to understand its limitations, and reasons why Dishonest Bob would prefer other attacks. The second example is of the most general attack, one in which the plaintext bits are probabilistically-related to the message bits. The last example is of a very special attack, one in which the plaintext bits are deterministically-related to the message bits. In section 4, we show that this type of attack is among the most efficient.

Example 1. The *repeated query/plaintext attack* consists of requesting the same bit n times, for odd n: $(t_i, t_i, t_i, ..t_i)$. Bob receives n randomized bits, $(\phi(t_i), \phi(t_i), \phi(t_i), ..\phi(t_i))$. If $\rho > \frac{1}{2}$, the estimated value of the single bit t_i is the bit most represented in the odd number of randomized responses. The efficiency of the attack, or its rate, R, is the number of message bits determined per plaintext bit, or $R = \frac{1}{n}$. It is well-known that this attack can be thwarted if Alice maintains a list of provided bits and never randomizes the same bit (or its complement) inconsistently. In our model of the protocol as a communication channel, requesting the same bits corresponds to using the repetition channel code. This code is known to be inefficient because decreasing the probability of estimation error requires that the rate, $\frac{1}{n}$, be decreased as well. In other words, $\omega_n \rightarrow 0$ implies $R \rightarrow 0$. Thus the repetition attack is not the best attack for two reasons: it is recognizable, and it is not particularly efficient.

Example 2. The *probabilistically-related plaintext attack.* Consider

t_1 “gender”

t_2 "Over 40"

x_1 "Losing Calcium"

x_2 "Balding"

x_3 "Greying"

x_4 "Gaining weight"

Suppose Bob wishes to determine the "message" t_1t_2, consisting of target bits t_1 and t_2. Assuming all messages are possible, the message values divide respondents into $M = 4$ categories. The number of plaintext bits is $n = 4$. The plaintext bits reveal information about the message, but they *do not determine it completely.* For example, women over 40 are more likely to be losing Calcium than any of the three other categories. Similarly, men over 40 are almost the only category balding. However, a man over 40 can have the same responses as a man under 40. We think of this as an attack sequence because the values of the string $x_1x_2x_3x_4$ are probabilistically related to the values of the message t_1t_2, and knowing the values of $x_1x_2x_3x_4$ decreases the error in estimating t_1t_2, though it may not reduce it to zero. The rate of the attack is defined as $R = \frac{log_2 M}{n} = \frac{1}{2}$.

Example 3. The *deterministically-related plaintext attack.* Consider a database of attributes of all residents of a county. Consider the set of bits:

x_1. "location = North";

x_2. "virus A test = positive";

x_3. "gender = male" AND "condition B = present".

Suppose it is also known that, for this county,

$$(location = North) \oplus (virusAtest = positive) \Leftrightarrow (gender = male) AND \quad (1)$$

$$(conditionB = present)$$

i.e,

$$x_1(\gamma) \oplus x_2(\gamma) = x_3(\gamma) \;\; \forall \; \gamma \in \Gamma \quad (2)$$

where $\oplus$ represents the XOR operation, γ an individual in the county, and Γ all individuals in the county. This could be determined, for example, from county health statistics.

Suppose Bob chooses as message bits x_1 and x_2 for all individuals. Suppose he wants to design an over-determined plaintext bit sequence by also requesting x_3. Without randomization, he would not need to do so. With randomization, x_3 serves as a parity check for the values of x_1 and x_2, or, in the communication channel framework, as an error-correcting symbol. The plaintext bits may be thought of as the code bits. We have chosen a simple example where the message bits form a subset of the plaintext bits, but this need not always be the case. In general, one can have an over-determined sequence of n plaintext bits whose values are completely determined by the message - through a set of n equations known to be satisfied by the messages and the plaintext bits. Equation (2) is one such equation.

The plaintext bits $x_1(\gamma)$, $x_2(\gamma)$, and $x_3(\gamma)$ form what we will define as deterministically-related plaintext (DRP). The value of the plaintext string is totally determined by the values of the message $(x_1(\gamma), x_2(\gamma))$. There exists a map, Λ, from the message to the string of plaintext bits, and corresponding maps from the message to each plaintext bit:

$$(x_1(\gamma), x_2(\gamma), x_3(\gamma)) = \Lambda(x_1(\gamma), x_2(\gamma)) = (x_1(\gamma), x_2(\gamma), x_1(\gamma) \oplus x_2(\gamma))$$

Assuming there are residents both with and without the virus in the North and in the South, no 2-tuple of message bits $(x_1(\gamma), x_2(\gamma))$ is *a priori* impossible, and the total number of messages is $M = 4$. The rate of the sequence is $\frac{log_2 M}{n} = \frac{2}{3}$.

If the attack is recognized, Alice could:

(a) refuse to respond

(b) respond with $\phi(x_1) \oplus \phi(x_2)$ instead of $\phi(x_1 \oplus x_2)$.

Recognizing the attack is not trivial. If, instead of "male with condition B", x_3 were, "$(location = North) \oplus (virusAtest = positive)$", it may be recognized by Alice, through extensive record keeping, as a logical combination of previously provided bits. But in the form of a request for a bit about gender and condition B, and in the absence of knowledge of the specific relationship of equation (1), or a causal relationship - as opposed to a statistical one in a limited population - gender and condition B are not readily seen to be revealing information regarding infection with virus A. Such an attack is fairly difficult to recognize, and hence to counter.

We use these examples of typical attacks to motivate a few definitions, which we present in the next section.

3 Definitions

In this section we define the three types of attacks. We also provide definitions of a channel and codes from information theory, which we shall need to prove our results. We provide a list of symbols in the appendix.

3.1 Attacks

An attack depends on the message bits that interest Bob, on the plaintext he uses to obtain his information, and on the way he estimates the message bits from the ciphertext.

Definition 1 A message m is a sequence of bits targeted by Bob, $\{t_i\}_{i=1}^{k}$, each bit being a combination of bits in the database, $t_i = h_i(\{a\}_{a \in A_y \subseteq \mathcal{D}})$, for boolean functions h_i, and database, or set of profiles, $\mathcal{D}$.

Definition 2 A *plaintext* p is a sequence of bits requested by Bob, $p = \{x_i\}_{i=1}^{n} = (r_i(\{a\}_{a \in A_y \subseteq \mathcal{D}}))_{i=1}^{n}$, for boolean functions r_i.

The plaintext is typically designed to distinguish among a pre-defined set of possible messages; each representing a distinct profile. The purpose of the plaintext is to determine which message is present, or what the values of the message bits are. For k message bits, the number of possible messages need not be 2^k, because certain message bit combinations may not be possible. Hence we define all plaintext string types and attacks in terms of the set of possible messages, $\mathcal{M}$, and its size, M. The value of $\mathcal{M}$, and hence of M, depends on Dishonest Bob's intentions, and typically cannot be determined from the plaintext. Knowing Dishonest Bob's intentions is not required to use our results, which provide bounds on what Dishonest Bob can do given his intentions.

Definition 3 An (M, n) plaintext for $\mathcal{M}$ is a plaintext of length n for the purpose of distinguishing among the M equal-length messages in $\mathcal{M}$. Its rate is $R = \frac{log_2 M}{n}$.

We require the messages to be equal-length because it simplifies the attack process. We show in Theorem 4 that this procedure provides a most efficient attack. An attack requires a plaintext and a way to estimate the message from the ciphertext, which, for want of a better term, we name *attack decoder.*

Definition 4 An *attack decoder* for protocol Φ, plaintext $p = \{x_i\}_{i=1}^{n}$ and message set $\mathcal{M}$ is an estimation function $\Psi : \Sigma^n \rightarrow \mathcal{M}$ for estimating the message from the ciphertext, $\{\phi(x_i)\}_{i=1}^{n}$.

Definition 5 An (M, n) *attack* on Φ using $\mathcal{M}$ consists of

(a) an (M, n) plaintext p for $\mathcal{M}$ and

(b) an attack decoder for Φ, p and $\mathcal{M}$.

The rate of the attack is the rate of the plaintext, $R = \frac{log_2 M}{n}$.

We now define the measure of the success of an attack, the probability of error.

Definition 6 The maximum probability of estimation error, ω_n, of an (M, n) attack on Φ using $\mathcal{M}$ is the maximum, over $\mathcal{M}$, of the probability of error for every $m \in \mathcal{M}$. The estimation error calculation assumes a "best" (maximum likelihood) estimation.

We now classify attacks based on the pattern among the plaintext bits. The most general attack, such as the one of Example 2, consists of a plaintext probabilistically-related to the message.

Definition 7 An (M, n) plaintext $p = \{x_i\}_{i=1}^n$ of $\mathcal{M}$, is a *Probabilistically-Related Plaintext (PP)* if the plaintext and the message are not independent, i.e. $I(m; p) \neq 0$.

$I(m; p)$ is the mutual information between the message and the plaintext, i.e. it is the change in uncertainty in m on knowing p. A PP is hence a plaintext string that reveals some information about the message. Notice that the definition assumes nothing about the relationship between plaintext bit x_i and previously received ciphertext bits: $\phi(x_1), \phi(x_2), ...\phi(x_{i-1})$. Hence, our definition most generally includes all types of plaintext attacks, including adaptive plaintext attacks. The attack of Example 3 is a special PP attack; the parity-check bit is not only highly correlated with the rest of the sequence, it is completely determined by it. We shall show later that such attacks are among the most efficient in reducing error.

Definition 8 An (M, n) plaintext p of $\mathcal{M}$ is a *Deterministically-related Plaintext (DP)* if there exists a DP map $\Lambda : \mathcal{M} \rightarrow \Sigma^n$; $\Lambda(m) \ = \ p$, mapping each message to an n-tuple of plaintext bits.

Clearly, a DP is a special PP, because $I(m; p) = \mathcal{H}(p) \neq 0$, i.e. the change in uncertainty in m on knowing p is not only non-zero, as required for a PP, but is the maximum possible. Our definition of DP does not consider the possibility of Bob designing his queries based on previous ciphertext. Hence we define the following:

Definition 9 An (M, n) plaintext p of $\mathcal{M}$ is a *Deterministically-related Adaptive Plaintext (DAP)* if there exists a set of DP maps $\Lambda_i : \mathcal{M} \times \Sigma^{i-1} \rightarrow \Sigma$; $\Lambda_i(m, \phi(x_1), \phi(x_2), ...\phi(x_{i-1})) \ = \ x_i$, mapping each message and possible previous ciphertext to a plaintext bit.

A DAP is clearly a PP. As the value of n for an attack grows, it is reasonable to expect the error to reduce, or, at least, not increase. We define attacks in which asymptotic error is zero as *small error* attacks.

Definition 10 A *small error attack* on Φ using the sequence of message sets $\{\mathcal{M}_n\}$ $|\mathcal{M}_n| = M_n$ is a sequence of (M_n, n) PP attacks such that $w_n \rightarrow 0$ as $n \rightarrow \infty$.

As a PP attack is the most general "main-channel" attack on the randomization protocol, a natural measure of the privacy of randomization would be the minimum plaintext length of a PP attack per bit of message entropy. In general, however, the plaintext length depends on the acceptable probability of estimation error. Motivated by the inefficiency of the small error repeated plaintext attack of Example 1, we define a reliable attack as one which can be made to decrease estimation error by increasing the total number of plaintext bits while maintaining rate; i.e., roughly, a small error attack of constant rate.

Definition 11 A *reliable attack of rate* R on Φ using the sequence of message sets $\{\mathcal{M}_n\}$, $|\mathcal{M}_n| = M_n$, is a sequence of $(2^{Rn}, n)$ attacks on Φ, using $\mathcal{M}_n$, such that $\omega_n \rightarrow 0$ as $n \rightarrow \infty$.

This definition draws from the definition of an "achievable" rate in information theory [9, pg. 194]. A reliable attack has quite a strong property: the maximum probability of estimation error can be made arbitrarily close to zero while maintaining the cost, per unit, of the attack. Such an attack provides a significant benefit to Dishonest Bob. Of course attacks that are not reliable would be of interest to him, but they would have a limitation: either rate would be traded off for accuracy, or the error would have a non-zero lower bound. The repeated query attack is clearly not reliable. In general, however, it is not necessary to have a reliable attack, because maximally efficient transmission for a given n might occur for a non-reliable attack. What is required is that the rate does not decrease to zero. Hence we define the following:

Definition 12 The *asymptotic rate* of a sequence of $(2^{R_n}, n)$ attacks on Φ is $Lim_{n\rightarrow\infty} R_n$.

A non-zero asymptotic rate corresponds to finite asymptotic plaintext length per bit of entropy, as we show in section 4.4.

Definition 13 The *privacy of randomization* is the (tight) asymptotic lower bound on the plaintext length, on average, per message, per bit of message entropy, in stationary message sequences, for a small error attack.

3.2 The protocol-channel

In this section we provide some definitions from information theory necessary for our results.

Definition 14 [9] A *communication channel* is a triplet of the following: a set of input variables, $\mathcal{X}$, a set of output variables, $\mathcal{Y}$, and a *a posteriori pdf*, $P(Y|X)$, and is denoted $(\mathcal{X}, P(Y|X), \mathcal{Y})$.

Trivially, the generalized oblivious transfer protocol is a communication channel with the additional condition that Alice is oblivious of the transfer details, though she is aware of the channel. We denote the channel corresponding to a protocol by Φ, and the channel corresponding to the binary symmetric protocol with probability of lie $1-\rho$ by $\Phi_{\mathcal{B}}(1-\rho)$.

Definition 15 The *channel capacity* of protocol Φ is the maximum decrease in entropy of variable X due to the protocol, and is denoted $\mathcal{C}(\Phi)$.

See Appendix for details about the definition. The channel capacity of the binary symmetric protocol with probability of a lie $1-\rho$ is

$$\mathcal{C}(\Phi_{\mathcal{B}}(1-\rho)) = 1 - \mathcal{H}(p) = 1 + \rho log_2 \rho + (1-\rho) log_2 (1-\rho)$$

bits, where $\mathcal{H}(p)$ is the entropy of the binary variable with p being the probability of one of its values. The channel capacity is essentially the "room" per bit afforded by the protocol to carry information about the plaintext. As described in section 1.2, it is similar to the *redundancy* in a language, defined in [24].

When the protocol has a small bias, i.e. $\rho = 0.5 \pm \beta$ for small β, its capacity is determined by the second order term of the Taylor expansion (zeroth and first order terms are zero):

$$\mathcal{C}(\Phi_{\mathcal{B}}(0.5 \pm \beta)) = \frac{2\beta^2}{ln2}, \beta \ small \tag{3}$$

Definition 16 [9, pg. 193] An (M, n) *binary channel code* for a binary channel $(\Sigma, P(Y|X), \Sigma)$ is a triplet $(\mathcal{M}, f, g)$ where (a)$\mathcal{M}$ is a domain of M messages, (b) f is an encoding function taking messages to codewords, $f : \mathcal{M} \rightarrow \Sigma^n$ and (c) g is a decoding function taking all possible channel output to messages, $g : \Sigma^n \rightarrow \mathcal{M}$. The rate of the code is $R = \frac{log_2 M}{n}$.

Definition 17 [9, pg. 213] An (M, n) *binary channel code with feedback* for a binary channel $(\Sigma, P(Y|X), \Sigma)$ is a triplet $(\mathcal{M}, \{f_i\}_{i=1}^n, g)$ where (a) $\mathcal{M}$ is a domain of M messages, (b) $\{f_i\}_{i=1}^n$ is a set of encoding functions each taking messages and channel output to code bits, $f_i : \mathcal{M} \times \Sigma^{i-1} \rightarrow \Sigma$; $f_i(m, \phi(x_1), \phi(x_2), ...\phi(x_{i-1})) = x_i$, and (c) g is a decoding function taking all possible channel output to messages, $g : \Sigma^n \rightarrow \mathcal{M}$. The rate of the code is $R = \frac{log_2 M}{n}$.

4 Our Results

We wish to demonstrate the existence of a tight asymptotic lower bound on the plaintext length, per message, per bit of message entropy, for zero asymptotic error. We do this by viewing the randomization protocol as a channel. Channel codes, because they increase the efficiency of data transmission over a channel, are attacks. In particular, we show a one-to-one correspondence between channel codes and DP attacks for a given message (i.e. channel codes are deterministically-related plaintext attacks). This correspondence maintains rate and estimation error. The channel coding ("Shannon's second") theorem [23, 9] shows a tight upper bound of channel capacity on the rate of a code if the transmission is to be reliable. This theorem thus says that reliable DP attacks exist for all rates below protocol capacity, and not for any rates larger than protocol capacity. In addition, by modifying the proof of the converse of the channel coding theorem using Fano's inequality [9, pg. 205], we show that channel capacity is also an upper bound on the rate of a reliable PP attack, and that it is an asymptotic upper bound on the rate of a small error PP attack. Thus our analogy with communication over a channel is as follows: the protocol is a channel, message bits a choice of source code, DP attacks channel codes, channel capacity the maximum rate of a reliable attack consisting of sequential queries, and Shannon codes most efficient reliable DP attacks. PP attacks do not correspond to channel codes, but Fano's inequality, the main ingredient for demonstrating channel capacity as a bound on the rate of a code, holds for a string of bits only probabilistically-related to the message. Thus Fano's inequality provides the *bound* on the rate of *all* main-channel small error attacks, and Shannon's proof of the existence of codes for all rates upto capacity provides the *existence* of DP (and hence PP) attacks for all rates upto capacity.

If the number of message bits used to represent a profile is its minimum value, the plaintext length of a maximum rate small error PP attack is this value divided by the channel capacity; using the source-channel coding theorem, we show that Bob cannot do better. This gives our final result, that the tight asymptotic lower bound on plaintext length for zero asymptotic error is the ratio of profile entropy to protocol channel capacity.

4.1 DP attacks and channel codes

We start by demonstrating a one-to-one correspondence between DP attacks and channel codes for a given set of messages. One direction for the correspondence is straightforward; a DP, being a sequence of queries on the messages, is also defined by a function on the message bits, which makes it a code. It is also easy to see that all codes are DP attacks; particularly when one observes that each code bit is defined by a boolean function on the message bits, themselves boolean functions on bits in the database.

Theorem 1 *Given a set of equal-length messages $\mathcal{M}$, and a randomization protocol Φ, a one-to-one correspondence exists between (a) the set of all $(|\mathcal{M}|, n)$ binary channel codes (with feedback) on set of messages $\mathcal{M}$, for channel Φ, and (b) the set of all $(|\mathcal{M}|, n)$ DP attacks (DAP attacks) on Φ using $\mathcal{M}$. The correspondence preserves rate and error.*

Proof. From Definitions 8 and 16, it is clear that an (M, n) DP attack on Φ using $\mathcal{M}$ corresponds to an (M, n) binary channel code for channel Φ, and message set $\mathcal{M}$, when $f = \Lambda$ and $g = \Psi$. Similarly, from Definitions 9 and 17, it is clear that an (M, n) DAP attack on Φ using $\mathcal{M}$ corresponds to an (M, n) binary channel code with feedback, for channel Φ, and message set $\mathcal{M}$, when $f_i = \Lambda_i$ and $g = \Psi$. The correspondence obviously preserves estimation error. Rate is preserved because values of M and n are preserved.

Given an (M, n) code, $\mathcal{C} = (\mathcal{M}, f, g)$, observe that, because f is a function from a message to a set of n bits, each of the n output bits is a boolean function on the bits of the message. Each bit of the message is itself a boolean function on bits in the database. More specifically, $f(m) = f(t_1, t_2, ...t_k) = (f_1(t_1, t_2, ...t_k), f_2(t_1, t_2, ...t_k), ...f_n(t_1, t_2, ...t_k))$ where $f_i : \Sigma^k \rightarrow \Sigma$. Substituting $t_i = h_i(\{a\}_{a \in A_{t_i} \subseteq \mathcal{D}})$, we get

$$f_i(t_1, t_2, ...t_k) = f_i(h_1(\{a\}_{a \in A_{t_1} \subseteq \mathcal{D}}), h_2(\{a\}_{a \in A_{t_2} \subseteq \mathcal{D}}), ...h_n(\{a\}_{a \in A_{t_k} \subseteq \mathcal{D}})) = r_i(\{a\}_{a \in A \subseteq \cup_i A_{t_i} \subseteq \mathcal{D}})$$

which satisfies the requirements of a plaintext bit (Definition 2), i.e. it is a boolean function on bits from the database. Hence, $f(m), m \in \mathcal{M}$ is the probabilistically-related plaintext of $\mathcal{M}$ corresponding to code $\mathcal{C}$, with $\Lambda = f$. The DP attack on Φ using $\mathcal{M}$ consists of the DP and the attack decoder $\Psi = g$. The correspondence clearly preserves rate and estimation error. Similarly for codes with feedback. □

This result can be used to obtain a large set of other results on the structure and existence of DP attacks. An interesting direction for further work is the type of structure imposed on DP attacks by certain types of channel codes, and possible implications for the recognizability/usability of such attacks.

4.2 The channel coding theorem and the existence of reliable DP attacks

Shannon's channel coding theorem demonstrates the existence of reliable codes for all rates lower than channel capacity. Using our definitions, all codes do not correspond to attacks; codes are defined on message sets that do not necessarily consist of equal-length messages, while attacks have been defined on equal-length messages. This is a trivial problem to get around - a code defined on a set of messages can be used to define another code on another set of the same size with the same rate and estimation error; we may choose the second set to consist of equal-length messages. Thus reliable codes acting on equal-length messages exist for all rates below channel capacity. Hence, using Theorem 1, DP attacks exist for all rates below protocol capacity.

The channel coding theorem [23], [9, pg. 198] *Reliable transmission is possible for all rates below capacity. Specifically, for every rate $R < \mathcal{C}$, there exists a sequence of $(2^{Rn}, n)$ codes with $\omega_n \rightarrow 0$.*

For a proof sketch, see Appendix. From the correspondence between channel codes for equal-length messages and DP attacks, the channel coding theorem applies immediately to DP attacks with the following simple observation:

Lemma 1 *If a sequence of $(2^{Rn}, n)$ codes with $\omega_n \rightarrow 0$ exists, such a sequence of codes exists for any sequence of messages, $\{\mathcal{M}'_n\}, |\mathcal{M}'_n| = 2^{Rn}$.*

Proof. Consider a sequence of $(2^{Rn}, n)$ codes $\mathcal{C}_n = (\mathcal{M}_n, f_n, g_n)$, such that $\omega_n \rightarrow 0$. Because $|\mathcal{M}_n| = |\mathcal{M}'_n|$, there exists a one-to-one function α_n, $\alpha_n : \mathcal{M}_n \rightarrow \mathcal{M}'_n$ for each n. The sequence of codes $\mathcal{C}'_n = (\mathcal{M}'_n, f\alpha_n^{-1}, \alpha_n g)$ has the same errors and rate as the sequence $\mathcal{C}_n$, and its maximum error also tends to zero. □

Theorem 2 *For a binary symmetric randomization protocol Φ, $\forall R < \mathcal{C}(\Phi)$, given a sequence of sets of equal-length messages, $\{\mathcal{M}_n\}$, $|\mathcal{M}_n| = 2^{Rn}$, there exists a reliable DP attack of rate R on Φ*

using $\{\mathcal{M}_n\}$.

Proof. From the channel coding theorem and Lemma 1, for every rate $R < \mathcal{C}(\Phi)$, there exists a sequence of $(2^{Rn}, n)$ codes acting on the set of messages $\{\mathcal{M}_n\}$ with $\omega_n \to 0$. From Theorem 1, this sequence corresponds to a sequence of DP attacks with identical error and acting on the same sequence of messages. □

Theorem 2 says that DP attacks in which the rate remains the same (but decrease in error is paid for by increase in plaintext length) exist if the rate is below the protocol capacity. It does not say anything about how the attacks will be constructed, and whether the encoding and decoding functions, Λ and ψ, are computationally feasible. Recall that encoding is performed by the database, or by Alice, and its complexity is measured by the number of logical operations performed to produce a plaintext bit from points in the database. Some results since Shannon's work help address the issue of feasibility and construction. Forney's work, originally published in [13] (out of print) - a short summary of which is accessible in [17, pg. 129] - says that Shannon codes that are encodable and decodable in polynomial time ($O(n^4)$) exist. This implies that reliable polynomial-time DP attacks exist. More recent work, that of Spielman, [25] shows how to construct linear time encodable and decodable codes that approach the channel coding theorem's limits. Thus, linear time encodable and decodable DP attacks with rates approaching protocol capacity, and arbitrarily low error, can be constructed. It is likely that attacks modelled on good, computationally feasible, error-correcting codes would consist of plaintext bits that are rather contrived combinations of message bits. It is not clear how easy it would be to recognize such requests. Recognizability constraints, ignored by us, could affect the existence result.

4.3 The converse of the channel coding theorem and bounds on the rates of reliable PP attacks

Bounds on the rates of reliable DP attacks follow from the converse of the channel coding theorem, which applies to DP attacks through the correspondence between DP attacks on a set of equal-length messages $\mathcal{M}$ and channel codes on message set $\mathcal{M}$. We modify the proofs of the converse of the channel coding theorem with and without feedback to show it applies to reliable PP attacks as well; a few of the essential steps in the proof of the converse of the channel coding theorem do not require the bits of the code to be functions of the message. Further, we use the approach of the converse of the channel coding theorem to show that any PP attack with error small enough must have maximum rate close enough to protocol capacity. This implies that the asymptotic rate of a small error attack is bound above by channel capacity.

Converse of the channel coding theorem [23], [9, pg. 198 and 213]: *Any sequence of* (M, n) *codes, with or without feedback, with maximum (or average) probability of error* $\to$ *0 as* $n \to \infty$

must have rate not greater than capacity.

For a proof sketch, see Appendix.

Theorem 3 *The rate of a reliable (M, n) PP attack on Φ is bounded above by $\mathcal{C}(\Phi)$. Further, the rate of any small error PP attack is asymptotically bounded above by $\mathcal{C}(\Phi)$.*

Proof. The proof is almost identical to the proof of the converse of the channel coding theorem [9], except for a change to incorporate the fact that, in a PP, queries are not necessarily a function of messages. Assume $\omega_n \to 0$ as $n \to \infty$, i.e. the attack is small error. Then the average probability of error, E_n, also $\to 0$ as $n \to \infty$. Let the rate of the attack using n queries be denoted R_n. Consider the case when the messages, m_n, are equally likely. Then,

$$log_2 M = nR_n = \mathcal{H}(m_n) = \mathcal{H}(m_n|\phi(x_1), \phi(x_2), ...\phi(x_n)) + I(m_n; \phi(x_1), \phi(x_2), ...\phi(x_n))$$

From equation (8.95) (Fano's inequality), [9, pg. 205], which does not require x_i to be a function of m_n,

$$\mathcal{H}(m_n|\phi(x_1), \phi(x_2), ...\phi(x_n)) \leq 1 + E_n nR_n$$

and hence,

$$nR_n \leq 1 + E_n nR_n + I(m_n; \phi(x_1), \phi(x_2), ...\phi(x_n)) \tag{4}$$

Further,

$$\begin{aligned} I(m_n; \phi(x_1), \phi(x_2), ...\phi(x_n)) &= \mathcal{H}(\phi(x_1), \phi(x_2), ...\phi(x_n)) - \mathcal{H}(\phi(x_1), \phi(x_2), ...\phi(x_n)|m_n) \\ &= \mathcal{H}(\phi(x_1), \phi(x_2), ...\phi(x_n)) - \textstyle\sum_i \mathcal{H}(\phi(x_i)|\phi(x_1), \phi(x_2), ...\phi(x_{i-1}), m_n) \\ &\leq \mathcal{H}(\phi(x_1), \phi(x_2), ...\phi(x_n)) - \textstyle\sum_i \mathcal{H}(\phi(x_i)|\phi(x_1), \phi(x_2), ...\phi(x_{i-1}), m_n, x_i) \\ &= \mathcal{H}(\phi(x_1), \phi(x_2), ...\phi(x_n)) - \textstyle\sum_i \mathcal{H}(\phi(x_i)|x_i) \\ &\leq \textstyle\sum_i \mathcal{H}(\phi(x_i)) - \sum_i \mathcal{H}(\phi(x_i)|x_i) \\ &= \textstyle\sum_i I(x_i; \phi(x_i)) \\ &\leq n\mathcal{C}(\Phi) \end{aligned}$$

From equation (4),

$$nR_n \leq 1 + E_n nR_n + n\mathcal{C}(\Phi)$$

Hence,

$$R_n \leq \frac{\frac{1}{n} + \mathcal{C}(\Phi)}{(1 - E_n)}$$

and

$$Lim_{n\to\infty} R_n \leq \mathcal{C}(\Phi)$$

□

Theorem 3 means that if it is required that the maximum error over a set of specified messages be made arbitrarily small on increasing the number of requests, the asymptotic rate of the attack cannot be larger than protocol capacity. If the length of the message is k, the rate of the attack is not $\frac{k}{n}$, but $\frac{log_2 M}{n}$ for M k-tuples of interest, where M need not be 2^k. Overestimating $log_2 M$ can lead to overestimating rate. One can then conclude that the rate of the attack is too large for the attack to be reliable over a particular channel when, in reality, this is not so. Consider the following example and circumstances where overestimating the rate is possible.

Dishonest Bob is using a DP of length n, concentrating on k-tuples of target bits for an entire population. Every k-tuple is possible. The rate would be $\frac{k}{n}$. He is interested in reducing maximum estimation error over each k-tuple to arbitrarily low values by increasing n and maintaining rate, i.e. he is interested in a reliable attack. This is only possible if the channel capacity is at least as large as $\frac{k}{n}$. This constraint on channel capacity is not always necessary when a reliable attack is desired, the length of the messages is k, and the number of plaintext bits n. To see this, consider the following:

Case 1. Dishonest Bob targets a few types of individuals, those who would respond in a certain way to queries on the target attributes. This limits the number of messages possible, i.e. $M < 2^k$, or $log_2 M < k$. The rate is $\frac{log_2 M}{n}$, which is strictly smaller than $\frac{k}{n}$. Hence reliable attacks of this kind, using k message bits at a time, can exist for channels with capacity not as large as $\frac{k}{n}$.

Case 2. Suppose Dishonest Bob is interested only in a subset of the message bits, in, say, $k_1 < k$ values. Again, the attack has rate lower than $\frac{k}{n}$.

Case 3. If some (other) bits of Alice's profile are completely determined from the message bits, and these other bits are of interest to Dishonest Bob, he could attach these to the message. This will increase the number of message bits, but will not affect the number of messages, hence it will not change rate if the number of plaintext bits is maintained. It is easy (and incorrect) to think that tagging k_1 bits to the length k message changes the rate to $\frac{k+k_1}{n}$, and that it changes certain small error attacks to not being small error on a particular channel because of the rate increase.

Typically, the rate of the attack is not apparent to Alice, because she is not even aware of the messages of interest to Dishonest Bob, she only knows the plaintext. Even if she were aware of the number of message bits, k, she would not be able to estimate the rate of an attack because of the various scenarios described above. What is clear to Alice is the probability of truth, and hence, the channel and its capacity. Given Theorem 3, Alice knows that she can choose channel properties, through a choice of the parameters of the probabilistic perturbation of her responses, to limit the asymptotic rate of a small error attack. She can think of a single plaintext bit as providing at most $\mathcal{C}(\Phi)$ bits of information.

Our definition of an (M, n) PP attack might be a bit confusing to the coding theorist - a $(2^k, n)$ code typically corresponds to n equations in k unknowns, and when a PP attack is not a DP attack, a coding theorist will be tempted to think of it as a code of rate unity. It *is* a DP (i.e a channel code) of rate unity, if message bits and plaintext bits are identical, but this is not the only situation in which it is an attack. It can be an attack for various kinds of message bits. In addition to DP attacks, where the coding theory analogy holds, we have tried to understand non-DP attacks, (i.e. other PP attacks) when there may be no equations at all among message bits and plaintext bits. For this situation there is no direct analogy with coding theory, and rate is not well-defined only by the plaintext.

Case 4. Consider a last example [22] of a set of n plaintext bits, each highly correlated with the first. Assume that none of the plaintext bits is completely determined by any of the others, i.e. $\mathcal{H}(x_i|x_1, x_2, ...x_{i-1}, x_{i+1}, ...x_n) \neq 0 \forall i$. Assume that each bit is also a message bit, i.e. that $x_i = t_i$. Assume also that, for a given value of n, all n-tuples of message bits are possible. However, as n increases, a message, or, equivalently, plaintext, is “typical”, i.e. most bits are equal to the first. As all bits are message bits, the attack is a $(2^n, n)$ DP attack of rate unity. The attack cannot be reliable for probabilities of a lie greater than zero (i.e for channel capacity lower than unity). However, Dishonest Bob can learn a lot from the ciphertext. Thus he can learn quite a bit about the message bits in cases where the rate of the attack is larger than the bounds of Theorems 3 and 4. What he cannot do is drive error to zero without reducing rate. If only the first bit were a message bit, this example would be a reasonable $(2, n)$ PP replacement for the repetition DP. It would have rate $\frac{1}{n}$, and there is a chance that it is not recognizable. It is clearly not reliable.

4.4 The source-channel coding theorem and the privacy of randomization

Theorem 3 provides a maximum rate of transfer of a given number of message bits over a channel of given capacity, i.e., given a number of message bits, it provides a minimum plaintext length (corresponding to maximum rate) if error is to be made arbitrarily small while maintaining rate. If $\mathcal{H}(\mathcal{P})$ is the entropy of the profile, using a proof almost exactly like that of the source-channel coding theorem [23, 9] (in fact, using the source-channel separation implicit in the proof), we show that a tight asymptotic lower bound on the number of queries required, on average, per profile, is $\frac{\mathcal{H}(\mathcal{M})}{\mathcal{C}(\Phi)}$ when the profile sequence is stationary, and that choosing messages independently of the pattern among plaintext bits is an optimal strategy. The source-channel coding theorem shows that the choice of the messages and the choice of the pattern among the plaintext can be made independently to yield optimal efficiency for stationary sources. Its statement, however, assumes a single source and a single channel, and that transmission is not backed up, i.e. the time rate at which bits are transmitted from the source is the same as the time rate at which the channel carries bits away. We do not need to assume a single channel. A single channel corresponds to a single plaintext bit per message. Hence we cannot use the source-channel coding as it is usually stated.

Theorem 4 *The tight asymptotic lower bound on the plaintext length, on average, per message, for a small error PP attack, is* $\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ *if the message sequence is stationary, i.e.*
$Lim_{n\to\infty}\omega_n \to 0 \Rightarrow Lim_{numberofmessages\to\infty}\frac{plaintextlength}{numberofmessages} \geq \frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$.

Proof.

$\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ is an asymptotic lower bound: Assume the existence of a small error attack with asymptotic plaintext length $K = \frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)} - \Delta$ per message on average, $\Delta > 0$. This means that, given $\epsilon, \delta > 0$, plaintext of size at most $n(K + \epsilon)$ for n messages, n large enough, can result in a probability of error at most δ. By Theorem 3, for any given ν, the rate of the attack can be at most $\mathcal{C}(\Phi) + \nu$, for large enough n, and hence the number of message bits at most $n(K+\epsilon)(\mathcal{C}(\Phi)+\nu) = n\mathcal{H}(P) - n\mathcal{C}(\Phi)(\Delta - \epsilon) + nK\nu + n\epsilon\nu$, i.e. each message is represented, on average, by a number of bits strictly smaller than the message entropy for small enough ϵ, δ, ν. This violates Shannon's source coding theorem [9, pg. 89, Thm. 5.4.2] and [23].

$\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ can be achieved from above (i.e. tightness): For a stationary message sequence and given ϵ and δ, it is possible to find a value of n such that n messages or more can be represented using a string of $n(\mathcal{H}(P)+\frac{\epsilon\mathcal{C}(\Phi)}{2})$ bits, with probability of representation error at most $\frac{\delta}{2}$ [9, pg. 52, equation 3.7]. Using a good code, it is possible to design a good DP attack with error at most $\frac{\delta}{2}$, of rate $\mathcal{C}(\Phi) - \frac{\epsilon\mathcal{C}(\Phi)}{2(\frac{\mathcal{H}}{\mathcal{C}(\Phi)}+\epsilon)}$, for n large enough. Thus, using a total of $\frac{n(\mathcal{H}(P)+\frac{\epsilon\mathcal{C}(\Phi)}{2})}{\mathcal{C}(\Phi)-\frac{\epsilon\mathcal{C}(\Phi)}{2(\frac{\mathcal{H}}{\mathcal{C}(\Phi)}+\epsilon)}} = n(\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)} + \epsilon)$ protocol instances, it is possible to estimate the n messages with probability of error at most $\frac{\delta}{2} + \frac{\delta}{2} = \delta$ for large enough n. That is, a reliable DP attack with $\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)} + \epsilon$ plaintext length per message exists. □

Corollary 1 *The tight asymptotic lower bound on the plaintext length, on average, per message, for a small error PP attack on* $\Phi_{\mathcal{B}}(0.5 \pm \beta)$ *is* $\frac{ln2\times\mathcal{H}(P)}{2\beta^2}$ *if the message sequence is stationary.*

Proof The result follows from Theorem 4 and equation (3). □

Corollary 2 *The tight upper bound on the plaintext length, on average, per message, for any probability of error using protocol* Φ*, is* $\frac{\mathcal{H}(P)}{\mathcal{C}(\Phi)}$ *if the message sequence is stationary.*

Proof. Follows from Theorem 4. □

Corollary 3 *The privacy of* Φ *is* $\frac{1}{\mathcal{C}(\Phi)}$.

Proof. Follows from Theorem 4 and Definition 13. □

Corollary 4 *The privacy of* $\Phi_{\mathcal{B}}(0.5 \pm \beta)$ *is* $O(\frac{1}{\beta^2})$.

Proof. Follows from Corollary 1 and Definition 13.

4.5 Our contributions in context

4.5.1 Complexity theory

One version of our results, that a lower bound on the number of points required to accurately estimate a single point for a small bias protocol is $O(\frac{1}{\beta^2})$, is not new. It follows from the results on "Chernoff-type bounds" [18, 16, 5, 8] - that the lower bound on the number of coin tosses required to determine accurately the probability $(0.5\pm\beta)$ of heads from a sequence of independent identically distributed tosses is $O(\frac{1}{\beta^2})$. For example, from the Chernoff bound:

$$n = \frac{[ln(\frac{2}{\delta})]}{0.38\beta^2} \Rightarrow Prob[error] \ \leq \ \delta$$

for the repetition code. While this bound, like ours, is $O(\frac{1}{\beta^2})$, it depends on the probability of error δ. We have shown that the number of points required per target point need not increase indefinitely, or at all, while decreasing error to zero - if one is willing to use a large enough number of points, and if one is willing to sample combinations of distinct target points. In other words, our bound is independent of δ. When the bias is not small, i.e. the channel capacity of the protocol is high, the difference between the Chernoff bound and the inverse of the channel capacity is significant.

Another related piece of work in complexity theory is on the problem of "twenty questions" with lies [12]. The work is a worst-case complexity analysis of the problem, with the errors appearing not at random, but in a worst-case distribution. Further, the model in this body of work does not allow the combination of data points in a single query, as is possible in our privacy model.

4.5.2 The theory of security

As mentioned earlier, our work provides a bound similar to the unicity distance of [24], and we provide a means of determining if our bound is tight, and of achieving it. Crepeau and Kilian [6] propose the use of ρ-noisy transfer as a primitive with which to build more powerful protocols such as oblivious transfer. Thus, their use of the noisy transfer protocol is to build (near) perfectly secret protocols.

Our result on the correspondence between channel codes and DP attacks is an example of the study of attacks on non-perfect protocols using results from coding theory. Interesting further results could follow from viewing non-perfect anonymous delivery protocols, such as Crowds and non-perfect combinations of mixes, as channels. Unlike cryptographic protocols, the *a posteriori* pdfs for these protocols can be explicitly expressed in simple form, and can be used to define channels, and thereafter, channel codes. Ramp secret sharing schemes, with explicitly defined *a posteriori* pdfs $(P(Y|X))$, are also amenable to this approach. An even more interesting direction of further work is to determine if our approach provides ingredients for a theory of statistical attacks.

4.5.3 Statistical measures of randomization

The database community has measures of the privacy of randomization [15, 3, 2]; these are, however, not motivated by a security analysis. [2] proposes the use of the differential mutual information between the original and perturbed continuous-valued data points as a measure of "conditional privacy loss", which inspires our measure. The mutual information between two variables, (see equation(5), Appendix), is the change in uncertainty of one on knowing the other. The measure of conditional privacy loss addresses the *change* due to a protocol instance. Because it is based on entropy, it also distinguishes among situations where the two possibilities are almost equally likely and situations where this is not so. It does, however, depend on the original pdf, and not only on protocol parameters. Our privacy measure, the inverse of the protocol channel capacity, is closely related to this measure, but improves on it by being independent of the input pdf (channel capacity is the maximum value of the mutual information, taken over all possible input pdfs).

In statistical databases, it is typically assumed that a larger number of queries (per attribute desired) is required for a lower error. Our proof of the existence of small error attacks for all asymptotic rates below channel capacity implies that a finite, fixed number of queries, per attribute desired, can ensure asymptotic error is zero. Further, our work demonstrates that there are attacks other than the repetition attack, that may not be as recognizable, and are less expensive per target attribute. Last, at first glance it might appear that combinations of a greater number of data points for a request provides greater protection of the data points. This is not true. A repetition attack is not reliable, but computationally feasible reliable DP attacks that reduce error arbitrarily can be constructed for rates below capacity. This implies that when one responds to a request that combines a large number of points, one may not be protecting the individual points; in fact one might easily be providing the computation for Dishonest Bob's error correcting codes.

5 Conclusions

We have approached the problem of privacy, in situations where there is benefit to providing information, as a game between Alice and Bob through the use of non-perfect protocols with controllable parameters. A "high privacy" equilibrium in such a world provides maximum benefit for minimum revealed information. We treat the binary symmetric randomization protocol as a channel, and channel codes as efficient attacks. We have demonstrated a number of interesting results, including (a) an equivalence between channel codes and DP attacks using a given set of messages, (b) existence results on reliable attacks of all rates below protocol channel capacity, (c) an asymptotic upper bound of protocol channel capacity on the rate of all small error attacks that use sequential queries, and (d) a lower bound on the plaintext length required per message per bit of message entropy, assuming the message sequence is stationary, and asymptotic error zero. We have defined a measure of variable privacy motivated by our results. We are not aware of any other work that uses the channel coding theorem or similar work to study the properties of any cryptographic protocols, nor of any other work that connects attacks on randomization to channel codes. Interesting open problems include (a) whether viewing all non-perfect protocols as channels, and efficient attacks as codes, can provide a theory of statistical attacks, and shed more light on the efficiency of attacks (even if non-polynomial-time) on other protocols; (b) optimal strategies for Alice and Bob in various common scenarios; (c) the use of variable privacy in the economic modelling of privacy.

6 Acknowledgements

We would like to thank Umesh Vazirani for the original suggestion to use randomization to study the market for privacy, and for encouraging the study of variable privacy. We would also like to thank him for spirited discussions, obstinate questioning of our view, and an observation enabling Theorem 1.

References

[1] Nabil R. Adam and John C. Worthmann, "Security-control methods for statistical databases: a comparative study", *ACM Computing Surveys*, Vol. 21, No. 4, pp. 515-556, December 1989.

[2] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms", *Proceedings of the Twenteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Santa Barbara, California, USA, May 21-23 2001.

[3] R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining", *Proc. of the ACM SIGMOD Conference on Management of Data*, Dallas, May 2000.

[4] Blakley, G.R., Meadows, C., "Security of ramp schemes", *Proc. of Crypto'84*, Lecture Notes on Comput. Sci., 196, pp. 242–268, Springer Verlag, 1984.

[5] Ziv Bar-Yossef, Ravi Kumar and D. Sivakumar, "Sampling Algorithms: Lower Bounds and Applications", *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC)*, pp. 266-275, 2001.

[6] Gilles Brassard and Claude Crepeau, "Oblivious transfers and privacy amplification", *Advances in cryptology: EUROCRYPT '97*, Lecture Notes in Computer Science, vol. 1223, pp. 334-347, 1997.

[7] Gilles Brassard, Claude Crepeau, Jean-Marc Robert, "Information-theoretic reductions among disclosure problems", *Proc. 27th IEEE Symposium on Foundations of Computer Science (FOCS)*, 1986.

[8] Ran Canetti, Guy Even and Oded Goldreich, "Lower Bounds for Sampling Algorithms for Estimating the Average", *Information Processing Letters*, Vol. 53, pp. 17-25, 1995

[9] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, John Wiley and Sons, 1991.

[10] Claude Crepeau and Joe Kilian, "Achieving oblivious transfer using weakened security assumptions", *29th Symposium on Foundations of Computer Science*, pp. 42–52, IEEE, 1988.

[11] Claudia Díaz, Stefaan Seys, Joris Claessens and Bart Preneel, "Towards measuring anonymity", *Proceeedings of the Workshop on Privacy Enhancing Technologies*, San Francisco, 14-15 April 2002

[12] Aditi Dhagat, Peter Gcs, Peter Winkler, "On playing 'Twenty Questions' with a liar" *Proceedings of the third annual ACM-SIAM Symposium on Discrete Algorithms*, 1992, Orlando, Florida.

[13] David G. Forney, *Concatenated Codes*, MIT Press, Cambridge, Mass., 1966.

[14] Cormac Herley, personal communication.

[15] Diane Lambert, "Measures of Disclosure Risk and Harm", *Journal of Official Statistics*, 9, pp. 313-331, 1993.

[16] Michael Luby, *Pseudorandomness and cryptographic applications*, Princeton Computer Science Notes, 1996.

[17] Robert J. McEliece, *The Theory of Information and Coding*, Cambridge University Press, 2002.

[18] Rajeev Motwani and Prabhakar Raghavan, *Randomized Algorithms*, pp. 67-73, Cambridge University Press, New York, NY, 1995.

[19] Michael K. Reiter and Aviel Rubin, "Crowds: Anonymity for Web Transactions", *ACM Transactions on Information and System Security*, Vol. 1, No. 1, pp. 66-92, November 1998.

[20] Paul Resnick and Hal R. Varian "Recommender systems", *Communications of the ACM*, Volume 40 Issue 3, March 1997.

[21] Andrei Serjantov and George Danezis, "Towards an Information Theoretic Metric for Anonymity", *Proceeding of the Workshop on Privacy Enhancing Technologies*, San Francisco, 14-15 April 2002

[22] Gadiel Seroussi, personal communication.

[23] Claude Shannon, "A mathematical theory of communication", *Bell Systems Technical Journal*, vol. 27, pp. 379-423, July 1948.

[24] Claude Shannon, "Communication Theory of Secrecy Systems", *Bell Systems Technical Journal*, vol. 28, pp. 657-715, 1949.

[25] Daniel A. Spielman, "Linear-time encodable and decodable error-correcting codes", *IEEE Transactions on Information Theory*, Vol 42, No 6, pp. 1723-1732, 1996.

[26] Douglas R. Stinson, *Cryptography Theory and Practice*, CRC Press, 1995.

[27] Sudan, Madhu, "Algorithmic issues in coding theory", *Conference on Foundations of Software Technology and Theoretical Computer Science*, Kharagpur, India, 1997.

[28] Poorvi Vora, "Randomization and the economic value of privacy", draft, 2003.

[29] Dominic Welsh, *Codes and Cryptography*, Oxford University Press, 1998.

[30] A. C. Yao, "Theory and Application of Trapdoor Functions", 23^rd *IEEE Symposium on Foundations of Computer Science*, pp. 80-91, Chicago, Illinois, 3-5 November 1982.

A Appendix A: List of Symbols

ρ	probability of truth
X	input to protocol/channel, plaintext bit
Y	output of protocol/channel, ciphertext bit
Σ	$\{0,1\}$
$P(Y\mid X)$	posterior pdf, (or *a posteriori* pdf) of protocol/channel
$\phi(X)$	randomized value of X
n	number of queries or length of plaintext
k	number of target bits or length of message
$\mathcal{M}$, $\mathcal{M}_n$	set of target strings or messages
M	number of messages
Φ	protocol/channel
$\mathcal{C}(\Phi)$	capacity of Φ
P	profile
$\mathcal{H}(P)$	entropy of P
$\Phi_{\mathcal{B}}$	binary protocol
β	small bias of a binary protocol
$\mathcal{D}$	database, or collection of profiles, or source of information
a	a bit in $\mathcal{D}$
t_i	a target bit or a message bit
m	a message
R	rate of an attack or a code
Λ	DP map
p	query sequence or plaintext string
A_x	subset of $\mathcal{D}$
Ψ	attack decoder
ω_n	maximum error of protocol using specified $\mathcal{M}$ and Ψ and plaintext of length n
$I(x;y)$	mutual information between x and y, or decrease in uncertainty of one on knowing the other
Λ_i	DAP map
f	an encoding function
g	a decoding function
f_i	an encoding function for a feedback code
E_n	average probability of error of protocol using $\mathcal{M}$ and Ψ and specified plaintext of length n
DP	deterministically-related plaintext
PP	probabilistically-related plaintext
DAP	deterministically-related adaptive plaintext

B Appendix B: Information Theory Review

We have used [9] extensively for our understanding of the results of information theory and coding, and have tried to use its notation as far as possible in this paper. Another paper on the privacy of randomization [2] has also proposed the use of most of the following definitions for the study of randomization.

B.1 Some basic definitions

The entropy of a discrete-valued random variable X, which takes on values $x \in \mathcal{X}$ is [9, Chp.2]:

$$\mathcal{H}(X) = -\sum_{x \in \mathcal{X}} p(x) log_2(p(x))$$

The entropy may be thought of as the uncertainty in random variable X, and is maximum when each value of X is equally likely [9, pp. 13]. Shannon's source coding ("first") theorem [23] states that the entropy of a random variable is the minimum number of bits required, on average, to represent the variable.

The conditional entropy of the discrete-valued random variable X with respect to discrete-valued random variable Y which takes on values $y \in \mathcal{Y}$ is:

$$\mathcal{H}(X|Y) = \sum_{y \in \mathcal{Y}} p(y) \mathcal{H}(X|Y = y)$$

It is the uncertainty in X on knowing Y, averaged over all possible values of Y.

The mutual information between discrete-valued variables X and Y, or the decrease in uncertainty of one on knowing the other, is defined as the difference between the original entropy and the conditional entropy:

$$I(X;Y) = \mathcal{H}(X) - \mathcal{H}(X|Y)$$

It can be shown to be symmetric [9]:

$$I(X;Y) = \mathcal{H}(X) - \mathcal{H}(X|Y) = \mathcal{H}(Y) - \mathcal{H}(Y|X) \tag{5}$$

It can also be shown that $\mathcal{H}(X|Y) \leq \mathcal{H}(X)$ with equality if and only if X and Y are independent [9, pp.27, Thm.2.6.5], or, equivalently, if and only if the protocol satisfies the definition of perfect secrecy [24, 26]. For discrete randomization, the protocol is perfect if the probability of truth is equal to the probability of a lie.

The mutual information between two discrete random variables is a finite-valued, continuous and concave function of the probability distributions of the two variables. Hence its maximum exists [9, pp.27,Thm.2.6.5]. The maximum is defined as:

$$\mathcal{C}(\phi) = \begin{matrix}\max \\ \text{p(x)}\end{matrix} \ \mathcal{H}(X) - \mathcal{H}(X|Y) \tag{6}$$

The maximum thus measures the maximum amount of information communicated by a channel in a single instance. It is denoted the channel capacity of the communication channel taking X to Y.

B.2 The channel coding theorem [23], [9, pg. 198]

The channel coding theorem: Reliable transmission is possible for all rates below capacity. Specifically, for every rate $R < \mathcal{C}$, there exists a sequence of $(2^{Rn}, n)$ codes with maximum probability of error → 0.

We briefly review a proof of the channel coding theorem [9]. It draws considerably from the following:

The Weak Law of Large Numbers: For n independent, identically distributed (i.i.d.) random variables, the sample mean tends to the expected value as $n \rightarrow \infty$.

The proof of the channel coding theorem involves using the channel many times so as to draw from the weak law of large numbers. The main part of the proof shows that the average probability of error over many, randomly-chosen, codes is small, and hence that at least one good code exists. Decoding is assumed to be maximum likelihood decoding, i.e. a channel output is decoded to a codeword that was most likely to have generated the output. For a typical code word, decoding error occurs when (a) the output is not likely to have been generated by the input; or, (b) many input codewords could have generated the same output codeword. Errors due to (a) tend to zero for large code words as a consequence of the weak law. Errors due to (b) are low when the code words are far enough from one another given the channel's error properties, i.e. when there are fewer than $2^{n\mathcal{C}}$ code words, i.e. when $log_2 M < n\mathcal{C}$, or $R < \mathcal{C}$.

B.3 The converse of the channel coding theorem [23], [9, pg. 198]

The converse of the channel coding theorem is:

Any sequence of (M, n) codes with maximum (or average) probability of error → 0 as $n \rightarrow \infty$ must have rate $R \leq \mathcal{C}$.

We briefly describe the salient properties of the proof. The entropy of the input per code bit when the input is uniformly distributed is R. It can also be shown that the entropy of the input per code bit is bounded above by the sum of the channel capacity and another term that → 0 as $n \rightarrow \infty$ if the probability of error → 0. Hence, R is bounded above by the channel capacity as $n \rightarrow \infty$ if the probability of error → 0.

While the converse is proved by looking at the case of uniformly distributed input values, it holds for all input distributions, because input entropy per code bit is not greater than R. Note that error correction can be performed for rates larger than capacity, but the error cannot be driven to zero.