The Power of Refresh: a Novel Mechanism for Securing Low Entropy PII

Yuqian Li*, Yang Liu*, Zhifang Liu*, Jiwei Huang*, Zhen Chen†
Department of Computer Science and Technology
Tsinghua University
Beijing 10084, China
*{*liyuqian79, lywander, chimneyliu, hjw217*}@gmail.com, †zhenchen@tsinghua.edu.cn*

***Abstract*—Deterministic encryption for low entropy personally identifiable information (PII) is vulnerable to dictionary attack. It is particularly so because of an expedient method to enumerate possible PII's plain text instead of all possible keys. Deterministic encryption, however, is indispensable in the generation of hash or index of PII.**

This paper presents a novel mechanism to frustrate dictionary attacks by refreshing the encryption in an external blackbox. The major part of this paper is about the analysis of this novel mechanism. The use of conditional entropy in this paper both measures the increased difficulty for attack and proves the value and feasibility of this novel mechanism. A lower bound for conditional entropy against a computationally-unbounded adversary is guaranteed. The essential meaning of the lower bound is also given based on min-entropy.

By the proof, this mechanism can provide very reliable security for PII in online social networks (OSN) and keep efficiency and functionality at the same time.

***Keywords*-personally identifiable information; online social network; conditional entropy; deterministic encryption; security.**

I. Introduction

Security of online personally identifiable information(PII) becomes an important concern [1], [2] because of the growing popularity of OSN such as Facebook[3] and the growing tendency for users to share their information on OSN. [4] explains the importance of protecting the confidentiality of PII and impacts of PII leakage in the context of information security. It also includes a list of confidentiality safeguards ranging from operational activities to technical methods.

Among those safeguards, some researches focus on how to minimize the use and collection of PII for third-party services. For example, [5] points out two defects in privacy policies of popular OSN and provides a method to find the minimum of private information needed for a particular set of interactions. In addition, [6] carries out a detailed analysis on possible ways of PII leakage via OSN. In those researches, the first-party (the party that directly serves the user, e.g. Facebook) server is considered to be safe to retrieve all PII data in OSN.

Many users, however, no longer trust the first-party server as more and more users become aware of the fact that security based on well performed management and shielded servers is not reliable since human mistakes, behavior of operators and server vulnerabilities are unpredictable.

To remove security dependence on the first-party, especially SNO(Social Network Operators), Buchegger et al.[7], [8] proposes the use of distributed social networks to control access. Though it achieves security independent of SNO, the traditional centralized server/client mode has so much advantage in simplicity and performance. Therefore, most popular OSN now still uses centralized mode and only controls PII access from normal users and third-party services.

Our work is to achieve the same level of security independent of SNO while preserving the simplicity and performance of the centralized mode. Moreover, two additional factors are also considered. Firstly, PII always has very low entropy for human memorization. Secondly, PII(e.g. cellphone numbers, email addresses) is always used as usernames for fast identifying and indexing in OSN. So our work has to guarantee the security of low entropy PII as well as functionality and efficiency of identifying and indexing based on PII.

Encryption of PII can be used for security but the encryption must be deterministic for fast identification and indexing. Deterministic encryption of low entropy PII, however, is vulnerable to dictionary attacks since the encryption system can hardly tell the difference between a normal human behaviour and a carefully designed automatic brute force dictionary attack. A dictionary attack can feasibly simulate normal human behaviour and enumerate all possible values of PII's plain text (this is only feasible for low entropy PII). Once the dictionary attack has enumerated all possible values and established a table which maps all cipher texts to their corresponding plain texts, the conditional entropy [9], [10] which measures uncertainty of the plain text under the condition of giving its corresponding cipher text and that table, is 0.

To ensure the security, this paper proposes a novel mechanism to guarantee a lower bound of conditional entropy for even computationally-unbounded adversaries. This mechanism is called defender mechanism since it's based on a defender model which introduces a defender to balance the adversary. In that mechanism, the defender simply refreshes the encryption in an external blackbox periodically. The analysis of the mechanism shows that a lower bound of conditional entropy can be guaranteed efficiently: suppose that the original entropy is E, a lower bound of conditional entropy $\Omega(E)$ can be guaranteed by performing one defend operation after $2^{\Omega(E)}$ possible attacks. Based on min-entropy, the proof shows that the expected chance to guess the right plain text is only $2^{-\Omega(E)}$ for a computationally-unbounded adversary.

The rest of this paper is organized as follows. Related work about entropic security is reviewed in section II. Section III introduces the defender model and mechanism. Section IV provides a proof as well as experiments. At last, we conclude the paper in section V.

II. Related Work

Entropy[11] measures the uncertainty of an information. Intuitively, it's easy to understand: the cipher text is regarded as secure if the adversary is uncertain about the plain text.

Entropic security is introduced by Russel and Wang[12] to define whether the cipher text leak any predicate of the plain text. However, it requires messages to have high entropy from the adversary's point of view. Similar entropic condition had been achieved in hash functions by Canetti et al[13], [14]. Hash functions are considered to be equivalent to deterministic encryption discussed in this paper: anyone could get a deterministic cipher text from a given plain text, but it's hard to find the corresponding plain text for a given cipher text. Entropic security has been further studied by [15] and its result applies to both encryption and hash functions.

However, all of their work only apply to high entropy messages. While for low entropy message, one simple contradiction has already been shown: enumerate all possible plain texts and a successful collision will recover the original plain text easily.

The key to this failure is the conditional entropy. Mutual information $I(X, Y)$ is widely used in the researches mentioned above. By definition,

$$I(X, Y) = H(X) - H(X|Y) \tag{1}$$

where $H(X)$ is the entropy of X and $H(X|Y)$ is the conditional entropy of X given Y (See definition 1, 3). In the previous work mentioned above, X is the plain text and Y is the cipher text. Whether $I(X, Y)$ is small is key to the security they defined. Note that $I(X, Y)$ is small is equivalent to that $H(X|Y)$ is large so $I(X, Y)$ itself does not have any problem. The problem is what Y is. If Y is only the cipher text, it's not suitable for the case that $H(X)$ is small. In the low entropy cases where the adversary could easily launch dictionary attacks, the information from these attacks is critical to the security. Therefore, Y should contain the information from those attacks for a better security analysis.

In summary, conditional entropy $H(X|Y)$ or equivalent mutual information $I(X, Y)$ has already been utilized in previous work. But in previous work, Y only contains the information of cipher text, while in this paper, Y is defined to contain the information gained by adversary during the dictionary attacks so a better analysis for low entropy encryption is achieved.

III. Defender Model and Mechanism

A. Defender Model

To frustrate dictionary attack, we propose a defender model which introduces a defender to balance the adversary who launches the attack.

The defender model is a simple model to intuitively describe the process about how the adversary (called attacker in this model) attacks step by step and how the defender is against the attacker correspondingly.

If only attacker exists, after each attack, some useful information is gained by the attacker. Once enough information is achieved by attacker, the security is compromised.

Initially, the attacker knows nothing. After each attack, the information gained is described as a function f_A, so define:

$$I_0 = 0 \tag{2}$$

$$I_r = f_A(I_{r-1}) \tag{3}$$

Here $I_r \in [0, 1]$ describes the amount of information gained after r attacks: 0 for nothing, 1 for enough information to compromise the security.

For attacker who enumerates all possibilities such as dictionary attacker, the function f_A is:

$$I_r = f_A(I_{r-1}) = I_{r-1} + c \tag{4}$$

where $c = 1/n$ is a constant depend on n, the number of possible instances to be enumerated. In high entropy situations, n is very large (e.g. greater than 2^{128}) so such an attacker needs too many attacks to compromise the security. Therefore, the system with a large n is secured if the attacker is computationally-bounded.

However, for low entropy PII, n is very small so a defender is introduced against that attacker. The defender's action is described as a function f_D so the equation(3) becomes:

$$I_r = f_A(f_D(I_{r-1})) \tag{5}$$

which means that defender will reduce the information gained by attacker before a new attack can be made.

For attackers who enumerate all possibilities, a defender who periodically reduces the information gained by attacker in a constant rate will be effective. With that defender, the equation becomes

$$I_r = f_A(f_D(I_{r-1})) = f_D(I_{r-1}) + c = I_{r-1}/d + c \tag{6}$$

Here $d > 1$ is the rate to reduce the information. It can be easily conducted that:

$$\lim_{r\to\infty} I_r = \lim_{r\to\infty} \sum_{0\le i<r} \frac{c}{d^i} = \frac{d \cdot c}{d-1}\ ,\ (d > 1)$$

This means that for a d such as 2, even c is as large as 0.1 (so without defender only 10 attacks will compromise the security) and the attacker is computationally-unbounded (as $r \to \infty$ says), security will never be compromised.

This model gives an abstract illustration about how to frustrate dictionary attacks for low entry PII. In the following sections, f_A will be replaced by a formal definition about dictionary attacks in a special system. In that system, a specially designed hardware (i.e. a blackbox) will be used to achieve f_D. And the information I will be formally measured by conditional entropy $H(X|Y)$: the greater $H(X|Y)$ is, the smaller I is.

B. Defender Mechanism

Based on defender model, a defender mechanism is implemented in a specific OSN which generates encrypted indexes from given PII. That PII is cellphone number in that specific OSN and the defender mechanism is to prevent attackers from knowing the corresponding cellphone number from its index. The defender mechanism also applies to other PII and cellphone numbers are introduced in this paper only for a concrete analysis and better understanding. In the specific OSN, indexes and corresponding encrypted user data entries are stored in server's database.

The defender mechanism is going to fulfil the defender function $f_D(I_{r-1}) = I_{r-1}/2$ by updating indexes. But If the attacker could track each new index from its old index, there's no loss of information for attacker so it has to make sure that the procedure to update the indexes is untrackable. This paper suggests a scenario that attackers(e.g. SNO) already escalated their privilege to root access for both server and database. Therefore, the system has to send indexes and data entries to another secured module and get new indexes and encrypted data entries from that module. The module is implemented in a special hardware (something like TPM, i.e. Thusted Platform Module) that nobody could see what's going on inside the hardware without damaging it in real world.

The best update strategy is to send all indexes and data entries to that hardware and then retrieve all new indexes and data entries together. By doing that, the attacker loses all information, which means $f_D(I_{r-1}) = 0$. However, the entries in database may be too many for that hardware to store. Therefore, in defender mechanism, all the n indexes and data entries are randomly partitioned into $n/2$ pairs. After the partition, two indexes and data entries in each pair are sent and retrieved between the database and hardware at a time. After that, the attacker can only guess which new index is from which old index from 2 possibilites. Thus, the equation $f_D(I_{r-1}) = I_{r-1}/2$ is fulfilled, intuitively.

The partition and update is enforced by the hardware so SNO and server can't bypass them. And in different updates, the random partitions are also different and independent.

In real system, doing such a defend operation to update whole database after each possible attack costs too much. Therefore, the defend operation is required after m possible attack operations rather than one. Under this new condition, f_D is unchanged, while $f_A(I_{r-1}) = I_{r-1} + c$ becomes $f_A(I_{r-1}) = I_{r-1} + c' = I_{r-1} + m \cdot c$. The m can be tuned in balance of security and the efficiency. The formal mathematical definition for defender mechanism's behavior will be given in the next section.

IV. Analysis of Conditional Entropy

This section gives out a proof and a demonstration of the defender mechanism's security under a computationally-unbounded attacker in information theory.

A. Definition and Example

In this subsection, a brief definition for entropy and conditional entropy is given along with the formal definition of defender mechanism and its behaviour. Some additional examples are taken to further demonstrate the relation between conditional entropy and the security.

Definition 1 (Entropy[11]): The entropy of a discrete random variable X with possible values $\{x_1, x_2, \ldots, x_n\}$, denoted as $H(X)$, is

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i) \tag{7}$$

where $\log$ refers to $\log_2$ in our context.

Correspondingly, in defender mechanism, define

Definition 2 (Attacker's Guess X): $\mathcal{X} = \{x_1, \ldots, x_n\}$ is the set of all possible primary images. For convenience, suppose that $n = 2^E$ (so the original entropy is E as illustrated below). The function $\varepsilon : \mathcal{X} \to Z$ is to generate index $z = \varepsilon(x)$ in a very large index space Z. For example, padding many deterministic bits to x and use a very long key to deterministic encrypt it. Considering all possible paddings and keys, Z is a very large space to adversaries. Suppose z^* is the index that attacker wants to get its primary image $x^* \in \mathcal{X}$ satisfying $\varepsilon(x^*) = z^*$. The discrete random variable X is the primary image guessed by the attacker against x^*.

In our context, primary images refer to PII. In our specific OSN, PII are cellphone numbers whose set $\mathcal{X}$ has a very small size about $10^{10} \approx 2^{32}$.

In ideal situation, the attacker has no related information, so X should be uniformly distributed:

$$H(X) = -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} = -\sum_{i=1}^{2^E} 2^{-E} \log 2^{-E} = E$$

Definition 3 (Conditional Entropy[9], [10]): For a discrete random variable X with possible values set $\mathcal{X}$ and another random variable Y with possible values set $\mathcal{Y}$, the conditional entropy of X given Y is

$$H(X|Y) = \sum_{y\in\mathcal{Y}} p(y)H(X|Y=y) \tag{8}$$

$$= -\sum_{y\in\mathcal{Y}} p(y) \sum_{x\in\mathcal{X}} p(x|y)\log p(x|y) \tag{9}$$

In our context, Y represents the related information achieved by attacker. In the special case that attacker has enumerated only one x to get $\varepsilon(x)$, $Y^{(1)}$ can be defined as:

$$\mathcal{Y}^{(1)} = \varepsilon(\mathcal{X}) = \{y \mid \exists x \in \mathcal{X}, \varepsilon(x) = y\}$$

$$\forall y \in \mathcal{Y}^{(1)},\ p(Y^{(1)} = y) = \frac{1}{|\mathcal{Y}^{(1)}|} = \frac{1}{n}$$

$$p(x|y) = \begin{cases} \frac{1}{n-1}, & y \neq z^* \\ 0, & y = z^* \text{ and } \varepsilon(x) \neq z^* \\ 1, & y = z^* \text{ and } \varepsilon(x) = z^* \end{cases}$$

Similarly, when attacker has enumerated m different primary images, $Y^{(m)}$ can be defined as:

$$\mathcal{Y}^{(m)} = \{y = \{y_1, y_2, \ldots, y_m\} \mid y_i \in \mathcal{Y}^{(1)}\}$$

$$\forall y \in \mathcal{Y}^{(m)},\ p(Y^{(m)} = y) = \frac{1}{|\mathcal{Y}^{(m)}|} = \frac{1}{\binom{n}{m}}$$

$$p(x|y) = \begin{cases} \frac{1}{n-m}, & z^* \notin y \\ 0, & z^* \in y \text{ and } \varepsilon(x) \neq z^* \\ 1, & z^* \in y \text{ and } \varepsilon(x) = z^* \end{cases}$$

Therefore, $H(X|Y^{(m)})$ can be calculated as:

$$\begin{aligned} H(X|Y^{(m)}) &= \sum_{z^*\in y} p(y)H(x|Y^{(m)} = y) + \\ &\quad \sum_{z^*\notin y} p(y)H(x|Y^{(m)} = y) \\ &= 0 + \frac{\binom{n-1}{m}}{\binom{n}{m}} \log(n-m) \\ &= \frac{n-m}{n}\log(n-m) \end{aligned}$$

In the special case of $Y^{(m)}$, as m increases, the conditional entropy decreases and drops to 0 when $m = n-1$. Consistent with the intuition, $H(X|Y^{(m)})$ which notifies the security decreases about linearly when m is small to n.

To make a simple proof of $H(X|Y)$ where Y contains all information achieved by attacker, we can simplify the condition Y and preserve the information it contains at the same time. Before defining Y, a formal definition of how defender mechanism behaviours is given. Clear definition of X can be found in definition 2 in the above.

The defender mechanism will allow at most m possible attacks (i.e. calculate m pairs of $(x, z = \varepsilon(x))$) before carrying out one defend operation. More specifically, between two operations of updating the indexes, at most m indexes are calculated from primary images(i.e. cellphone numbers). It's formally defined as:

Definition 4 (Defender Mechanism): There are functions $\varepsilon_0, \varepsilon_{-1}, \varepsilon_{-2}, \ldots$ where ε_0 denotes the most recent function $\varepsilon : \mathcal{X} \to Z$ to generate an index $\varepsilon(x) = z \in Z$ from primary image x. Besides, ε_{-1} denotes the last one used before update, ε_{-2} for the last but one and so on. For each ε_{-i}, at most m indexes $\varepsilon_{-i}(x)$ can be calculated. There are also functions $g_0, g_{-1}, g_{-2}, \ldots$ where g_{-i} is an update function $g_{-i} : Z \to Z$ such that $\varepsilon_{-i}(x) = g_{-i}\left(\varepsilon_{-(i+1)}(x)\right)$. Considering the capability of hardware, in ith recent update, n primary images in $\mathcal{X}$ are partitioned into $n/2$ pairs randomly:

$$P_{-i,j} = \{x_1, x_2\}\ ,\ (0 \le j < n/2)$$

The $g_i\left(\varepsilon_{-(i+1)}(P_{-i,j} = \{x_1, x_2\})\right)$ for one pair:

$$g_i\left(\varepsilon_{-(i+1)}(P_{-i,j})\right) = \{g_i\left(\varepsilon_{-(i+1)}(x_1)\right), g_i\left(\varepsilon_{-(i+1)}(x_2)\right)\}$$

is calculated in the hardware at a time. All indexes are updated by using g_i to all $n/2$ pairs $P_{-i,j}$. Since the hardware is a blackbox, the attacker can only track $\varepsilon_{-(i+1)}(P_{-i,j})$ and

$$\varepsilon_{-i}(P_{-i,j}) = g_{-i}\left(\varepsilon_{-(i+1)}(P_{-i,j})\right) = \{z_1, z_2\}$$

for each $P_{-i,j}$. But the attacker don't know whether $g_{-i}(\varepsilon_{-(i+1)}(x_1)) = z_1$ or $g_{-i}(\varepsilon_{-(i+1)}(x_1)) = z_2$. Here, $P_{-i,j}$ are totally random to the attacker and independent for different i.

To better describe the interaction among attacks and defends, candidate sets and collision sets are defined. The candidate set C_{-i} can be described as the set of indexes calculated by ε_{-i} that are possible to be updated to z^* through $g_0, g_{-1}, \ldots, g_{-(i-1)}$. The collision set K_{-i} is the subset of C_{-i} that is enumerated by the attacker.

Definition 5 (Candidate and Collision Set): Candidate sets are $C_0, C_{-1}, C_{-2}, \ldots$ recursively defined as

$$\begin{aligned} C_0 &= \{z^*\} \\ C_{-(i+1)} &= \{z \mid \exists P_{-i,j},\ g_{-i}(z) \in \varepsilon_{-i}(P_{-i,j}) \text{ and} \\ &\qquad \varepsilon_{-i}(P_{-i,j}) \cap C_{-i} \neq \emptyset\} \end{aligned}$$

Here z^* is the index that attacker wants to know its primary image x^* such that $\varepsilon_0(x^*) = z^*$. And collision sets are $K_0, K_{-1}, K_{-2}, \ldots$ where

$$K_{-i} = \{z \mid \varepsilon_{-i}(x) = z \text{ is calculated and } z \in C_{-i}\}$$

Note that the greater $|C_{-i}|$ is and the smaller $|K_{-i}|$ is, the more uncertain x^* is for attacker given this certain condition. In our measurement of conditional entropy $H(X|Y)$, Y is also a random variable, which means the condition itself is uncertain. $H(X|Y)$ is an average of entropy with certain condition $H(X|Y = y)$ over the distribution of $p(Y = y)$.

Now define condition Y_r that contains all information attacker achieved during infinite attacks. Here subscript r is a parameter to simplify condition's definition:

Definition 6 (Simple Condition Y_r): Y_r is a random variable with values set $\mathcal{Y} = \{\alpha, \beta\}$ where $Y_r = \alpha$ means that $|C_{-r}| = 2^r$ and $|K_{-i}| = 0$ $(0 \le i < r)$. Otherwise $Y_r = \beta$. Here r is a parameter to denote an event that the defender mechanism is lucky for r updates. Here lucky means that the best condition for defender is achieved.

B. Proof of a Lower Bound

Now we will prove a lower bound of $H(X|Y_r)$. Note that Y_r contains all information that attacker can achieve, so a lower bound of $H(X|Y_r)$ should also be a lower bound of any $H(X|Y)$ where Y is a condition denotes any information achieved by attacker under defender mechanism. In short, the lower bound of $H(X|Y)$ should be a maximum over all lower bound of $H(X|Y_r)$:

Theorem 1: Under defender mechanism, the primary image's conditional entropy has a lower bound of

$$H(X|Y) \ge \max_r \{L(E, m, r)\}$$

given all the information that a computationally-unbounded attacker can have in an infinite long time.

Here $L(E, m, r)$ is the lower bound of $H(X|Y_r)$ where m is the maximum number of indexes that are allowed to be calculated between defend operations and $E = \log(n)$ denotes the logarithm of the size of primary image set.

To prove the lower bound of $H(X|Y_r)$, use the definition of conditional entropy,

$$\begin{aligned} H(X|Y_r) &= p(Y_r = \alpha)H(X|\alpha) + p(Y_r = \beta)H(X|\beta) \\ &\ge p(Y_r = \alpha)H(X|\alpha) + 0 \end{aligned}$$

Here, the second part is just counted as 0 since it's non-negative. In other words, to get the lower bound of the conditional entropy $H(X|Y_r)$, an average of entropy $H(X|Y_r = y)$ over all possible conditions $y \in \mathcal{Y}$, any $H(X|Y_r = y)$ is counted as minimum value 0 except for the best condition $Y_r = \alpha$.

To calculate $p(Y_r = \alpha)H(X|\alpha)$, three lemmas are proposed. The first one shows a lower bound of $H(X|Y_r = \alpha)$ and the other two for a lower bound of $p(Y_r = \alpha)$. The lower bound of $H(X|Y_r)$ is achieved by putting them together.

Lemma 1 ($H(X|Y_r = \alpha)$'s Lower Bound):

$$H(X|Y_r = \alpha) \ge r \cdot (1 - \frac{rm \cdot 2^r}{2^E - 2^r + 1}) \ , \ (2^r < 2^E - rm)$$

Proof: When $Y_r = \alpha$, the attacker is unlucky in last r updates and defender is lucky to expand C_{-r} quickly. In this case, the best that attacker could still know are all relations like $\varepsilon_{-r}(x) = z$ for all $x \in \mathcal{X}$.

In addition, $H(A) \ge H(A|B)$ for any A, B, which simply means that knowing something more can never be a bad thing. Let $A = (X|Y_r = \alpha)$ and B be whether there is any $\varepsilon_{-r}(x) \in C_{-r}$ whose x has been enumerated using $\varepsilon_0, \varepsilon_{-1}, \ldots, \varepsilon_{-(r-1)}$ by the attacker or not. Then

$$\begin{aligned} H(X|Y_r = \alpha) &\ge H(A|B) \\ &\ge p(B = false) \cdot H(A|B = false) \end{aligned}$$

When $B = false$, each $\varepsilon_{-r}(x_i) = z_i \in C_{-r}$ has an equal chance of $\varepsilon_0(x_i) = z^*$. Therefore:

$$H(A|B = false) \ge -\sum_{i=1}^{2^r} \frac{1}{2^r} \log(\frac{1}{2^r}) = r$$

For $p(B = false)$, use simple counting method:

$$\begin{aligned} p(B = false) &= \frac{\binom{n-rm}{2^r}}{\binom{n}{2^r}} \\ &= \frac{(n - rm) \ldots (n - 2^r - rm + 1)}{n(n-1) \ldots (n - 2^r + 1)} \\ &\ge (\frac{n - 2^r - rm + 1}{n - 2^r + 1})^{2^r} \\ &= (1 - \frac{rm}{n - 2^r + 1})^{2^r} \\ &\ge 1 - \frac{rm \cdot 2^r}{2^E - 2^r + 1} \end{aligned}$$

So finally:

$$\begin{aligned} H(X|Y_r = \alpha) &\ge p(B = false) \cdot H(A|B = false) \\ &= r \cdot (1 - \frac{rm \cdot 2^r}{2^E - 2^r + 1}) \end{aligned}$$

■

To prove a lower bound of $p(Y_r = \alpha)$, note that $(Y_r = \alpha)$ is equivalent to $(|C_{-r}| = 2^r$ and $|K_{-i}| = 0$ $(0 \le i < r))$. Therefore

$$\begin{aligned} p(Y_r = \alpha) &= p(|C_{-r}| = 2^r) \\ &\quad \cdot p(|K_i| = 0 \ (0 \le i < r) \setminus |C_{-r}| = 2^r) \end{aligned}$$

For convenience, define

$$\begin{aligned} p_2 &= p(|C_{-r}| = 2^r) \\ p_3 &= p(|K_i| = 0 \ (0 \le i < r) \setminus |C_{-r}| = 2^r) \end{aligned}$$

And for the convenience of the following subsection, define $p(B = failse) = p_1$.

The following two lemmas are for p_2 and p_3 respectively.

Lemma 2 (Lower Bound of p_2):

$$p_2 = p(|C_{-r}| = 2^r) \ge 1 - \frac{r \cdot 2^{2r-2} + r \cdot 2^{r-1}}{2^E - 1}$$

Proof: $|C_{-r}| = 2^r$ means that $\varepsilon_{-i}(P_{-i,j}) \cap C_{-i} \le 1$ for all $P_{-i,j}$ $(i \le r - 1)$. Define

$$p(D_i) = p\left((\forall P_{-i,j}, P_{-i,j} \cap C_{-i} \le 1) \setminus |C_{-i}| = 2^i\right)$$

So

$$p(|C_{-r}| = 2^r) = \prod_{i=0}^{r-1} p(D_i)$$

It's obvious that $\forall i < r, p(D_i) \geq p(D_{r-1})$, therefore

$$p(|C_{-r}| = 2^r) \geq p(D_{r-1})^r = P^r$$

P here can be easily estimated by counting method as

$$\begin{aligned} P &= \frac{(n-2^{r-1})\ldots(n-2^r+1)\cdot(n-2^r-1)!!}{(n-1)!!} \\ &= \frac{(n-2^{r-1})(n-2^{r-1}-1)\ldots(n-2^r+1)}{(n-1)(n-3)\ldots(n-2^r+1)} \end{aligned}$$

Since

$$\frac{n-2^{r-1}-i}{n-1-2i} \geq \frac{n-2^{r-1}-j}{n-1-2j} \text{ when } i \geq j$$

It can be conducted that

$$\begin{aligned} P &= \frac{(n-2^{r-1})(n-2^{r-1}-1)\ldots(n-2^r+1)}{(n-1)(n-3)\ldots(n-2^r+1)} \\ &\geq (\frac{n-2^{r-1}}{n-1})^{2^{r-1}} \end{aligned}$$

Thus

$$\begin{aligned} p(|C_{-r}| = 2^r) &\geq P^r \\ &\geq (\frac{n-2^{r-1}}{n-1})^{r\cdot 2^{r-1}} \\ &= (1-\frac{2^{r-1}+1}{n-1})^{r\cdot 2^{r-1}} \\ &\geq 1-\frac{r\cdot 2^{2r-2}+r\cdot 2^{r-1}}{2^E-1} \end{aligned}$$

■

Lemma 3 (Lower Bound of p_3):

$$\begin{aligned} p_3 &= p(|K_{-i}| = 0\ (0 \leq i < r) \setminus |C_{-r}| = 2^{-r}) \\ &\geq 1-\frac{2^{r-1}mr}{2^E-2^{r-1}+1} \end{aligned}$$

Proof:

$$\begin{aligned} & p(|K_{-i}| = 0\ (0 \leq i < r) \setminus |C_{-r}| = 2^{-r}) \\ & \geq p(|K_{r-1}| = 0 \setminus |C_{r-1}| = 2^{r-1})^r \\ & = \left(\frac{\binom{n-2^{r-1}}{m}}{\binom{n}{m}}\right)^r \\ & \geq \left(1-\frac{2^{r-1}m}{n-m+1}\right)^r \\ & \qquad \text{(see proof of lemma1 for similar conclusion)} \\ & = 1-\frac{2^{r-1}mr}{n-m+1} \end{aligned}$$

■

By putting them together, here comes the lower bound

$$\begin{aligned} H(X|Y_r) &\geq p(Y_r = \alpha)\cdot H(X|Y_r = \alpha) \\ &= p_2\cdot p_3\cdot p_1\cdot r \\ &\geq (1-\frac{r\cdot 2^{2r-2}+r\cdot 2^{r-1}}{2^E-1}) \\ &\quad \cdot(1-\frac{2^{r-1}mr}{2^E-m+1})\cdot(1-\frac{rm\cdot 2^r}{2^E-2^r+1})\cdot r \\ &= L(E,m,r) \end{aligned}$$

The formula above is a little complex. A much easier asymptotic result could be derived from that. Suppose that $r = c_1E, m = 2^{c_2E}$ $(c_1, c_2 < 1)$ and E is large enough:

$$\begin{aligned} H(X|Y_r) &= L(E,m,r) \\ &= \left(1-\frac{c_1\cdot E}{2^{E-2c_1E+2}}+o(1)\right) \\ &\quad \cdot\left(1-\frac{c_1E}{2^{E-c_1E+1-c_2E}}+o(1)\right) \\ &\quad \cdot\left(1-\frac{c_1E}{2^{E-2c_2E}}+o(1)\right)\cdot c_1E \end{aligned}$$

Therefore, when $2c_1, c_1+c_2, 2c_2 < 1$, for example $c_1 = c_2 = 1/3$, and E is large enough, there exists:

$$m = 2^{c_2E} = 2^{\Omega(E)}$$
$$H(X|Y_r) = c_1E + o(E) = \Omega(E)$$

So the following theorem is derived:

Theorem 2: Under defender mechanism, primary image's conditional entropy has a lower bound of $\Omega(E)$ given all the information that a computationally-unbounded attacker can have in an infinite long time when one defend operation is enforced after $2^{\Omega(E)}$ calculations of indexes. Here $E = \log(n)$ is the logarithm of the size of primary image set.

C. Analysis of the Special Case

In our specific OSN dealing with cellphone numbers, $E = 32$. The simple lower bound proved above when $m = 2^{11}, 2^{12}, 2^{13}, 2^{14}$ is given in figure 1.

Since our lower bound in theorem 1 is a maximum value over r, the x-axis is r and the peak of each line is the lower bound for each m. As it shows, when $m = 2^{13} = 8192$, the lower bound is about 10.3. Thus only one update operation after thousands of index calculations is required to guarantee a lower bound higher than 10.

In fact, the proved lower bound in theorem 1 is so simple and the tight lower bound is expected to be much higher for the following reasons. Observing the proof of theorem 1, only the entropy in the situation $Y_r = \alpha$ is count. However, in many situations that $Y_r = \beta$, there is still a high entropy. What's more, $B = false$(see proof of lemma 1) is also assumed so the attacker is given an extra information about whether all his enumerated x in recent r updates are in

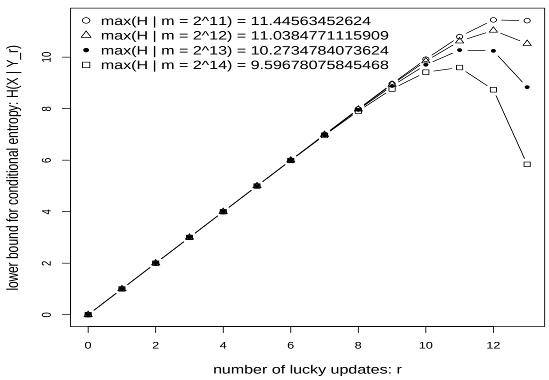

Figure 1. lower bound proved for different m

candidate set C_{-r}. But in real situation, this is unknown to the attacker.

D. Experimental Evaluation and Essential Meaning

In the this subsection, experiments are conducted to evaluate the result and an essential meaning based on min-entropy of our lower bound is given.

In our proof above, p_1, p_2 and p_3 are three key points to the final result and each of them has a lemma to prove its lower bound. And the lower bound $L(E, m, r)$ of $H(X|Y_r)$ is achieved by:

$$H(X|Y_r) \geq p_1 \cdot p_2 \cdot p_3 \cdot r \geq L(E, m, r)$$

In fact, $p_1 \cdot p_2 \cdot p_3$ can be measured in a real program which simulates the same behaviour as defender mechanism. So the proof above can be evaluated by this experiment. Moreover, this experiment will show how tight our lower bound is when $H(X|Y_r = \beta)$ and $H(A|B = true)$ are ignored.

The experiment program simply simulates the whole process of r updates for 10000 times and records the number of successful events to estimate the possibility $p_1 \cdot p_2 \cdot p_3$.

Figure 2 show $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{14}$ and figure 3 show the conditional entropy's lower bound based on $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{14}$.

It can be seen that the simple lower bound proved is valid and not too far away from the experimental result. After all, the experiment could derive a little more tight lower bound: it is greater than 10 when $m = 2^{14} = 16384$.

Now, we will figure out some essential meaning of our proof. All the analysis above is based on Shannon entropy. Min-entropy $H_\infty(X)$ define the entropy in a new way:

$$H_\infty(X) = \min_{x \in \mathcal{X}} \left(-\log p\left(X = x\right)\right)$$

Similar conditional min-entropy could also be defined.

The simple proof above also applies to this min-entropy because $p(x|Y = y)$ is either 0 or 2^{-r} which implies:

$$-log(0) = \infty > -log(2^{-r}) = r$$

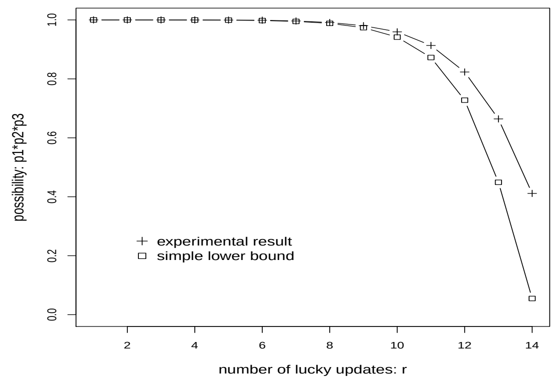

Figure 2. $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{14}$

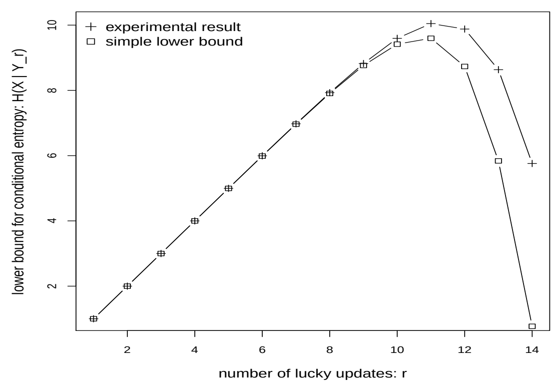

Figure 3. conditional entropy's lower bound when $m = 2^{14}$

And the lower bound for Shannon entropy and min-entropy in our proof is the same for the same reason. As it can be seen that min-entropy's definition looks simpler, it's easier to find out the meaning of the lower bound for conditional min-entropy. For min-entropy with a deterministic condition, $H_\infty(X|Y = y) = a$ denotes the highest probability 2^{-a} that adversary can achieve to guess the right answer when $Y = y$ is known. Since $H_\infty(X|Y) = E\left(H_\infty(X|Y = y)\right)$, conditional min-entropy means the expected highest possibility that one adversary can guess the right answer.

Therefore, here is the essential meaning of our proof: the proof of our lower bound shows that the expected highest possibility that a computationally-unbounded adversary can guess the right plain text is very small, i.e. $2^{-\Omega(E)}$. And it's 2^{-10} in the special case for $E = 32$ and $m = 2^{14}$.

V. Conclusion

To ensure the security of deterministic encryption for low entropy PII, this paper presents a novel defender model as well as a defender mechanism which has already been implemented for a specific OSN which uses cellphone numbers to generate encrypted indexes.

This paper mainly focuses on analysis of this defender mechanism. A lower bound of conditional entropy is calculated to prove the mechanism's security for even computationally-unbounded adversaries while the system's efficiency is also kept. Asymptotically, suppose that the original entropy is E, a lower bound for conditional entropy of $\Omega(E)$ can be guaranteed when only one defend operation is required after $2^{\Omega(E)}$ attacks. Based on min-entropy, our proof shows that such an adversary only has an expected chance less than $2^{-\Omega(E)}$ to guess the right plain text. However, the tight lower bound is believed to be much higher than we proved. In short, it's theoretically secured and should be more practically secured.

The defender mechanism also applies to other low entropy PII and will help other similar OSN to secure PII such as usernames. Meanwhile it can keep the functionality and efficiency of using that PII. By giving the theoretical proof in this paper, this mechanism should be much more reliable than shielded servers and well performed management which often have unpredictable vulnerabilities and mistakes.

ACKNOWLEDGMENT

This work is supported by Natural Science Foundation of China No. 90718040, National High-Tech Program No.2007AA01Z468, and Hosun Tech. The authors would also like to thank all those professors, teaching assistants and students who helped improving our project and reviewing this paper. Especially thanks to professor Xiaoge Wang from Tsinghua University and student Zhiyu Shang from Northwestern University.

REFERENCES

[1] A. Rabkin, "Personal knowledge questions for fallback authentication: security questions in the era of facebook," in *SOUPS '08: Proceedings of the 4th symposium on Usable privacy and security*. New York, NY, USA: ACM, 2008, pp. 13–23.

[2] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Commun. ACM*, vol. 50, no. 10, pp. 94–100, 2007.

[3] "Facebook." [Online]. Available: http://www.facebook.com/

[4] E. McCallister, T. Grance, and K. Scarfone, "Guide to protecting the confidentiality of personally identifiable information (pii)," 2010. [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-122/sp800-122.pdf

[5] B. Krishnamurthy and C. E. Wills, "Characterizing privacy in online social networks."

[6] B. Krishnamurthy and C. E. Wills, "On the leakage of personally identifiable information via online social networks."

[7] S. Buchegger and A. Datta, "A case for p2p infrastructure for social networks - opportunities & challenges," in *WONS'09: Proceedings of the Sixth international conference on Wireless On-Demand Network Systems and Services*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 149–156.

[8] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta, "Peerson: P2p social networking: early experiences and insights," in *SNS '09: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*. New York, NY, USA: ACM, 2009, pp. 46–52.

[9] G. A. Korn and T. M. Korn, *Mathematical Handbook for Scientists and Engineers: Definitions, Theorems, and Formulas for Reference and Review*. New York: Dover, 2000.

[10] C. Arndt, *Information Measures: Information and its description in Science and Engineering*. Berlin: Springer, 2001.

[11] C. E. Shannon, "Mathematical handbook for scientists and engineers: Definitions, theorems, and formulas for reference and review," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 5, no. 1, pp. 3–55, 2001.

[12] A. Russel, S. Key, E. Russell, and H. Wang, "How to fool an unbounded adversary with a short key," 2002.

[13] R. Canetti, "Towards realizing random oracles: Hash functions that hide all partial information." Springer-Verlag, 1997, pp. 455–469.

[14] R. Canetti, D. Micciancio, and O. Reingold, "Perfectly one-way probabilistic hash functions."

[15] Y. Dodis, "Entropic security and the encryption of high entropy messages," in *In Theory of Cryptography Conference (TCC) 05*. Springer-Verlag, 2005, pp. 556–577.