Anti Brute Force on Very Low Entropy Deterministic Encryption

Authors Name/s per 1st Affiliation (Author)
line 1 (of Affiliation): dept. name of organization
line 2: name of organization, acronyms acceptable
line 3: City, Country
line 4: Email: name@xyz.com

Authors Name/s per 2nd Affiliation (Author)
line 1 (of Affiliation): dept. name of organization
line 2: name of organization, acronyms acceptable
line 3: City, Country
line 4: Email: name@xyz.com

***Abstract*—Some information has a very small domain set. In that case, any deterministic encryption could only achieve a very low entropy bounded by the size of domain set. However, the deterministic encryption is required in situations such as hashing or generating an index or identity for a given information.**

For such very low entropy encryption, the adversary does not need to attempt every possible key. Instead, a successful brute force attack to enumerate every possible plain text and check whether the cipher text is the same would be easy. A secured system is designed and implemented to defend that attack. That secured system is called defender system for the reason that it's based on a simple idea called defender model from us.

To ensure the security in a theoretical aspect for such cases, this paper uses conditional entropy instead of mere entropy. The conditional entropy measures the difficulty for an adversary to derive the information considering all related information that adversary has already achieved. In this system, a relatively high lower bound for conditional entropy can be guaranteed even with a computationally-unbounded adversary who launches brute force attacks as above. In this paper, a very easy proof and some other approximation and experiment are given to show the lower bound. The practical meaning of conditional entropy's lower bound is also given from the aspect of min-entropy.

***Keywords*-[TODO]component; formatting; style; styling;**

I. Introduction

Deterministic encryption for an information from a very small domain set is important for some social network systems such as LiveS Cube. LiveS Cube[1] is a system to build a social network based on the address books in cellphones. In that social network, each node is indexed by a cipher text generated from the cellphone number of its user. Therefore, a huge network could be easily established using existing address books in numerous of cellphones without any additional effort of the users. However, cellphone numbers are one of the most important privacy[ref] so the security of preventing someone getting the corresponding cellphone number from its index is the base of the system.

But the security of such deterministic encryption is hard to be guaranteed, especially on a level that is not dependent on social network operators[ref]. It's true that the encryption could use a very strong key so that it's almost impossible to enumerate the right key to convert cipher text to plain text. However, the function to get the corresponding cipher text from a given plain text must be permitted for all social network operators and normal users, otherwise the social network system could not operate. Therefore, brute force attack can enumerate all possible values from the small domain set and establish a table maps any cipher text to its plain text as long as the encryption does not change. In such case, the conditional entropy [1], [2] given that table is 0 since there's no uncertainty for the adversary to get the plain text. Even for a large domain set which has a quite high entropy, a computationally-unbounded adversary could establish that table and the conditional entropy drops to 0 again. So it's not the mere entropy [1] that determines the security, but the conditional entropy considering all the information that adversary could gain during attacks. The related entropic security analysis will be reviewed in section II.

Therefore, even for the very low entropy information, there is still possibility to ensure the theoretical security by giving a lower bound for conditional entropy considering all information gained by possible brute force attacks from adversary. The defender model is an easy model based on a simple idea to achieve that goal. This model is used in LiveS Cube system so a defender system is made. The analysis in the following sections is mainly based on that system. The result shows that the system could ensure a relatively high lower bound of conditional entropy in a very efficient way: suppose that the original entropy is D, a lower bound of $c_1 D = \Omega(D)$ can be guaranteed by performing one defend operation after $2^{c_2 D} = 2^{\Omega(D)}$ brute force attacks. Since the analysis is quite simple, the bound estimated is believed not to be so tight. Considering the result derived by some other more complex, however not accurate, only approximate ways, it is believed that the tight lower bound for this system is much higher than what we proved. Experiments are also conducted to evaluate the lower bound. They confirm that our lower bound is valid but far from tight in some cases. At last, a practical meaning of our conditional entropy's lower bound is given from the aspect of min-entropy: the adversary is expected to compromise the

[1]this is a system still under developing

security in a chance of only $2^{-\Omega(D)}$.

In sum, our contribution is threefold. Firstly, to solve the security problem of low entropy information's deterministic encryption, defender model and system are proposed. Secondly, conditional entropy is introduced to analysis the theoretical security of system under computationally-unbounded brute force attacks and a simple lower bound of conditional entropy is proved in defender system. Thirdly, approximation and experiments are conducted to further check and evaluate our analysis and a practical meaning of our conditional entropy based proof is derived from the experiment in the aspect of min-entropy.

In the following sections, firstly some related works about privacy problems in social network systems and entropic security are reviewed. The defender model and system will be introduced after that. Then a simple proof for the lower bound of conditional entropy is given. Finally some other approximate and experimental ways to estimate the conditional entropy are discussed.

II. Related Work

A. Entropy and Entropic Security

Entropy[1] measures the uncertainty of an information. Intuitively, it's easy to understand how it could define the security: the more uncertain an adversary is about the information, the more security it has.

Entropic security is introduced by Russel and Wang[2] to define whether the cipher text leak any predicate of the plain text. However, it has to require the distribution on messages has high entropy from the adversary's point of view. Similar entropic condition had been achieved in hash functions by Canetti et al[3], [4]. Hash functions are considered to be equivalent as deterministic encryption discussed in this paper: anyone could easily 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. The only difference is that no key exists in hash functions and nobody could recover that plain text, while in our deterministic encryption, a very strong key exists and anybody who does not have that key could not recover that plain text. Since they key is considered unknown to adversaries and it's considered to be infinitely strong, deterministic encryption and hash functions become equivalent in the sense of security in our context.

Entropic security has been further studied by [5] and its result applies to both encryption and hash functions. However, all their work only apply to high entropy messages. While for low entropy message, their results seems not work. One simple contradiction for hash functions has already been shown: enumerate all possible plain text and see which causes a 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 those previous researches. 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[2]. In their researches, X is the plain text and Y is the cipher text. Therefore, whether $I(X,Y)$ is large is the key to the security they defined. In traditional work, $I(X,Y)$ is required to be very small, for example 0, so Y does not leak much information. And $I(X,Y)$ 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 mere cipher text, it's meaningless when the $H(X)$ is small. So even in the work of [5] where $I(X,Y)$ is allowed to be large, the same problem exists because Y is still not changed. In the low entropy cases, the adversary could launch simple brute force attacks, the information from these attacks is critical to the security. Therefore, Y should contain the information from those attacks for the security analysis. Thus conditional entropy $H(X|Y)$ for such Y could better describe security during the attacks. One simple conclusion is that when $H(X|Y)$ drops to 0, the adversary has compromised the security successfully because no uncertainty exists considering all the information Y after his or her attacks.

In sum, conditional entropy $H(X|Y)$ or similar mutual information $I(X,Y)$ has already been utilized in previous work. But in this paper, Y is defined to contain the information gained by adversary during the brute force attacks so a better analysis for low entropy encryption is achieved.

B. Privacy Problems in Online Social Network

Though defender system is currently used for only one specific LiveS Cube System, it is believed to be useful for many other social network systems for the following reasons.

Firstly, in recent years, OSNs like Facebook is becoming more and more popular, so that users put a lot of their private information on the Internet, which leads to serious security problems [6], [7]. Meanwhile, more and more social network systems begin to notice that security based on well performed administrators and unhackable servers is not reliable since human mistakes, behavior of operators and server vulnerabilities are unpredictable. So hardly can anyone still totally trust servers and administrators. Distributed social networks have been proposed by Buchegger et al.[8], [9] to ensure distributed access control. In [10] a new architecture is proposed to remove dependence on both the SNO(Social Network Operators) and other users, while preserving the simplicity and performance of traditional centralized server/client model. In this paper, a novel system based on centralized server/clients is proposed to make a social network with low entropy index (e.g. Phone numbers) secure and trusted in terms of both SNO and other users.

Secondly, not only password is worth being protected, but identity is also worth being protected as so many new social networks emerge every day. [Guide to Protecting...]

[2]See definition 1, 3

explains the importance of protecting the confidentiality of PII (Personally Identifiable Information, e.g., user ID) 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 these safeguards, some researches focus on how to minimize the use, collection and retention of PII. For example, [characterizing privacy] points out two defects in privacy policies of popular OSNs and provides an approach to evaluating the bare minimum of private information needed for a particular set of interactions. In addition, [On the leakage] carries out a detailed analysis on possible ways of PII leakage via OSNs. Our work, in contrast, is based mainly on two of the methods listed in [Guide to Protecting], i.e. de-identifying information and anonymizing information. In this way, prevention of leakage is no longer necessary since neither third-party services nor the first-party server is able to retrieve plain or recoverable PII data, even in ideally unbounded-computational brute force attack, while the basic function of PII, i.e. identification, is remained.

III. Defender Model and System

A. Defender Model

Defender model is a very simple model that is not based on formal information theory. The adversary is called attacker in this model. The attacker can launch some attacks and after each attack, the attacker gains some useful information. Once the attacker gains enough information, the security is compromised.

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

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

$$I_n = f_A(I_{n-1}) \tag{3}$$

Here $I_i \in [0, 1]$ describes the amount of information to compromise the security after i rounds of attack: 0 for nothing, 1 for enough information to compromise the security.

For a simple attacker who enumerates all possibilities, the function f_A is quite simple:

$$I_n = f_A(I_{n-1}) = I_{n-1} + c \tag{4}$$

Here c is a constant depend on how many possibilities the attacker has to enumerate. For example, if there are m possibilities, $c = 1/m$. In normal situations, m is more than 2^{128}, so such a simple attacker just needs too many rounds of attack to compromise the security. Therefore, the system with a large m is safe if the attacker is computationally-bounded.

However, in some situations, the m is very small. In such cases, we must introduce a defender against that attacker to ensure the security. The defender's action is also described as a function f_D so the equation(3) above becomes:

$$I_n = f_A(f_D(I_{n-1})) \tag{5}$$

For simple attacker who enumerate all possibilities, there is a simple but effective defender who periodically reduces the information attacker has in a constant rate, so the equation(4) becomes

$$I_n = f_D(I_{n-1}) + c = I_{n-1}/d + c \tag{6}$$

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

$$\begin{aligned} \lim_{n\to\infty} I_n &= \lim_{n\to\infty} \sum_{0\le i<n} \frac{c}{d^i} \\ &= \frac{c}{d-1} \end{aligned}$$

This means that for a small d such as 2, even c is as large as 0.1 and the attacker is computationally-unbounded, the attacker could never compromise the security.

B. Defender System

Inspired by the simple defender model, the defender system is implemented in LiveS Cube system which has to generate an index from a given cellphone number. The defender system is to prevent attackers from knowing the corresponding cellphone number from its index.

Define the set of cellphone number as $\mathcal{X} = \{x | x \text{ is a cellphone number}\}$. There's a function $f : \mathcal{X} \to F$ to generate index $h = f(x) \in F$ for a given x. As described before, the cellphone number set $\mathcal{X}$ has a very small size about $2^{32} \approx 10^{11}$. For a given index h^*, a simple attacker can enumerate all possible x to see whether $f(x) = h^*$. Therefore, in defender model:

$$f_A(I_{n-1}) = I_{n-1} + c = I_{n-1} + 2^{-32}$$

So the security will be compromised after 2^{32} rounds of attack if there's no defender. The defender system is going to fullfil the defender function $f_D(I_{n-1}) = I_{n-1}/2$ by generating new indexes. In LiveS Cube system, the indexes and corresponding data entries are stored in database which we think attacker may have an access to. To generate new indexes, the system has to send indexes and data entries to a secured module and get new indexes and data entries from that module. If attacker could track each new index and its old index, there's no loss of information for attacker. Therefore, we have to make sure that the attacker couldn't track the procedure to update the indexes. To do that, the module is implemented in a special hardware that nobody could see what's going on inside the hardware without damaging it in real world. The best strategy againt the attacker 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

$f_D(I_{n-1}) = 0$. However, the entries in database may be too many for that hardware to store. Therefore, the system sends two indexes and data entries from the database to hardware in one time and then retrieve their new indexes and data entries. After that, the attacker can only guess which new index is from which old index and there are 2 possibilites. Thus, the defender system fullfils the equation $f_D(I_{n-1}) = I_{n-1}/2$.

In short, in defender system, there's a function $g : F \to F$ regenerating new indexes from old indexes. The defender system will randomly choose $h_1, h_2 \in F$ that have not been regenerated yet in each time and generate $h'_1, h'_2 \in F$ in a way that attacker can't tell whether $h'_1 = g(h_1), h'_2 = g(h_2)$ or $h'_2 = g(h_1), h'_1 = g(h_2)$. By doing that, the information like $f(x) = h$ becomes information that there's $1/2$ chance $f'(x) = h'_1$ and $1/2$ chance $f'(x) = h'_2$. Thus $f_D(I_{n-1}) = I_{n-1}/2$ is achieved.

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 $f_A(I_{n-1}) = I_{n-1} + c$ becomes $f_A(I_{n-1}) = I_{n-1} + c' = I_{n-1} + m \cdot c$. The m can be tuned in balance of security and the efficiency of system.

The conclusions about the defender model and system above are not strictly proved because defender model itself is not well defined in theoretical aspect. However this is a comprehensive explanation to show how and why defender system works. In the following section, a mathematical proof will be given based on information theory to demonstrate that the defender system can truly ensure the security even for a computationally-unbounded attacker, consistent with what is claimed here.

IV. Analysis of Conditional Entropy

A. Definition and Examples

In this subsection, a brief definition for entropy and conditional entropy is given. Besides, the formal definition of defender system and its behaviour is given. Some additional examples are taken to further demonstrate the relation between conditional entropy and the security. Feel free to skip the content about the definition of entropy and conditional entropy if you are familiar with them.

Definition 1 (Entropy): The entropy of a discrete random variable X with possible values $\{x_1, x_2, \ldots, x_n\}$ 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 system, define

Definition 2: $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ is the set of all possible primary images[3] that index $h_i = f(x_i)$ can be calculated. Suppose h^* is the index that attacker wants to get its primary image $x^* \in \mathcal{X}$ satisfying $f(x^*) = h^*$. The discrete random variable X is the primary image guessed by the attacker. For convenience, suppose $n = 2^D$.

In ideal situation, the attacker has no related information, so X should be uniformly distributed. In such case, the entropy is simply:

$$\begin{aligned} H(X) &= -\sum_{i=1}^{2^D} 2^{-D} \log 2^{-D} \\ &= D \end{aligned}$$

Definition 3 (Conditional Entropy): For a discrete random variable X with possible values set $\mathcal{X}$, suppose that there is another random variable Y with possible values set $\mathcal{Y}$, the conditional entropy of X given Y is

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

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

$$= -\sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p(x, y) \log p(x|y) \tag{10}$$

In our context, random variable Y represents the related information that attacker has. For example, when attacker has enumerated only one x to calculate $f(x)$, Y can be defined as $Y^{(1)}$:

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

$$\forall y \in \mathcal{Y}^{(1)},\ p(Y^{(1)} = y) = \frac{1}{n} = \frac{1}{2^D}$$

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

Similarly, when attacker has enumerated m different primary images, Y can be defined as:

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

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

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

[3] the primary images are cellphone numbers in LiveS Cube system

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

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

It's clear that as m increases, the conditional entropy decreases and drops to 0 when $m = n - 1$. Consistent with the intuition, the conditional entropy which notifies the security decreases about linearly when m is small compared with n.

B. Proof of a Simple Lower Bound

As the example above demonstrates, the key to the calculation of conditional entropy $H(X|Y)$ is the definition of the condition Y. To make a simple proof for the lower bound, we can simplify the condition to achieve that. Before defining Y, let's more clearly clarify how defender system behaviours first. Clear definition of X can be found in definition 2 if that is unclear.

The defender system will allow attacker to do at most m possible attack operations before one defend operation. More specifically, between two operations of updating the whole database about the indexes and data entries, at most m indexes are calculated from primary indexes. It's formally defined as:

Definition 4 (Defender System): There are functions $f_0, f_1, f_2, \ldots$ where f_0 denotes the most recent function $f : \mathcal{X} \to F$ to generate an index $f(x) = h \in F$ from primary image x. Besides, f_1 denotes the last one used before update, f_2 for the last but one and so on. There are also functions $g_0, g_1, g_2, \ldots$ where g_i is an update function $g_i : F \to F$ such that $f_i(x) = g_i(f_{i+1}(x))$. Considering the capability of hardware and security, g_i is designed in a way that:

$$\begin{aligned} \{g_i(f_{i+1}(x_1)), g_i(f_{i+1}(x_2))\} &= \{f_i(x_1), f_i(x_2)\} \\ &= G_i(x_1, x_2) \end{aligned}$$

But attacker don't know whether $g_i(f_{i+1}(x_1)) = f_i(x_1)$ or $g_i(f_{i+1}(x_1)) = x_2$. Here, set G_i is defined for convenience in later proof and it's also totally random to the attacker. [4]

To better describe the interaction between attacker and defender system, candidate sets and collision sets is defined as following

Definition 5 (Candidate and Collision Set): Candidate sets are $C_0, C_1, C_2, \ldots$ recursively defined as

$$\begin{aligned} C_0 &= \{h^*\} \\ C_i &= \{h | \exists G_{i-1}(x_1, x_2), g_{i-1}(h) \in G_{i-1}(x_1, x_2) \text{ and} \\ &\qquad G_{i-1}(x_1, x_2) \cap C_{i-1} \neq \emptyset\} \ \ (i \geq 1) \end{aligned}$$

Here h^* is the index that attacker wants to know its primary image x such that $f_0(x) = h^*$. And collision sets are $K_0, K_1, K_2, \ldots$ where

$$K_i = \{h | f_i(x) = h \text{ is enumerated by attacker and } h \in C_i\}$$

The candidate set C_i can be described as the set of indexes of f_i that could be updated to h^* through $g_0, g_1, \ldots, g_{i-1}$. The collision set K_i is the subset of C_i that are enumerated by the attacker. Since g_i and f_i is random to attacker and a maximum of m indexes are allowed to be calculated using f_i, the random distribution of $|K_i|$ is only related to $|C_i|$ and has a maximum of m.

Now condition Y will be simply defined as following:

Definition 6 (Simple Condition Y_d): Y_d is a random variable with values set $\mathcal{Y} = \{\alpha, \beta\}$ where $Y_d = \alpha$ means that $|C_d| = 2^d$ and $|K_0| = |K_1| = \ldots = |K_{d-1}| = 0$. Otherwise $Y_d = \beta$.

By the definition of conditional entropy,

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

To prove a simple lower bound, three lemmas are proposed. The first one shows a lower bound for $H(X|Y = \alpha)$ and the other two prove a lower bound for $p(Y = \alpha)$. The final lower bound of $H(X|Y)$ will be achieved by putting them together.

Lemma 1: [5] $H(X|Y_d = \alpha) \geq d \cdot (1 - \frac{dm^2}{2^D - m + 1})$

Proof: When $Y_d = \alpha$, the attacker is just unlucky in last d updates and our defender system is lucky to expand C_d quickly. In this case, the best that attacker may have is to know all relations like $f_d(x) = h$ for $x \in \mathcal{X}$.

In addition, $H(A) \geq H(A|B)$ for any A, B, which simply means that knowing something more can never be a bad thing. Let $A = X|Y_d = \alpha$ and B be whether there is any $x \in C_d$ that has been enumerated using $f_0, f_1, \ldots, f_{d-1}$ by the attacker or not. Then

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

When $B = false$, each $f_d(x_i) = h_i \in C_d$ has an equal

[4] See section III-B for the purpose of g

[5] See definition 2 for D

chance of $f_0(x_i) = h^*$. Therefore we have:

$$\begin{aligned} H(A|B = false) &\geq -\sum_{i=1}^{2^d} \frac{1}{2^d} \log(\frac{1}{2^d}) \\ &= d \end{aligned}$$

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

$$\begin{aligned} p(B = false) &= \frac{\binom{n-dm}{m}}{\binom{n}{m}} \\ &= \frac{(n-dm)\dots(n-dm-m+1)}{n(n-1)\dots(n-m+1)} \\ &\geq (\frac{n-dm-m+1}{n-m+1})^m \\ &= (1 - \frac{dm}{n-m+1})^m \\ &\geq 1 - \frac{dm^2}{n-m+1} \end{aligned}$$

So finally:

$$\begin{aligned} H(X|Y_d = \alpha) &\geq p(B = false) \cdot H(A|B = false) \\ &\geq d \cdot (1 - \frac{dm^2}{n-m+1}) \\ &= d \cdot (1 - \frac{dm^2}{2^D - m + 1}) \end{aligned}$$

■

To prove a lower bound of $p(Y = \alpha)$, the following fact is used. $(Y = \alpha)$ is equivalent to $\big(|C_d| = 2^d$ and $|K_i| = 0\ (0 \leq i < d)\big)$. Therefore

$$\begin{aligned} p(Y = \alpha) = p(&|C_d| = 2^d) \\ &\cdot p(|K_i| = 0\ (0 \leq i < d) \setminus |C_d| = 2^d) \end{aligned}$$

The following two lemmas are for $p(|C_d| = 2^d)$ and $p(|K_i| = 0\ (0 \leq i < d) \setminus |C_d| = 2^d)$ respectively.

Lemma 2:

$$p(|C_d| = 2^d) \geq 1 - \frac{d \cdot 2^{2d-2} + d \cdot 2^{d-1}}{2^D - 1}$$

Proof: $|C_d| = 2^d$ means that $G_i(x_1, x_2) \cap C_i \leq 1$ for all $G_i (i \leq d-1)$. Define

$$p(A_i) = p\left((\forall G_i(x_1, x_2), G_i(x_1, x_2) \cap C_i \leq 1) \setminus |C_i| = 2^i\right)$$

So

$$p(|C_d| = 2^d) = \prod_{i=0}^{d-1} p(A_i)$$

It's obvious that $\forall i < d, p(A_i) \geq p(A_{d-1})$, therefore

$$\begin{aligned} &p(|C_d| = 2^d) \\ &\geq p(A_{d-1})^d \\ &= P^d \end{aligned}$$

P here can be easily estimated by counting method as

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

Since

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

It can be conducted that

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

Thus

$$\begin{aligned} p(|C_d| = 2^d) &\geq P^d \\ &\geq (\frac{n-2^{d-1}}{n-1})^{d \cdot 2^{d-1}} \\ &= (1 - \frac{2^{d-1}+1}{n-1})^{d \cdot 2^{d-1}} \\ &\geq 1 - \frac{d \cdot 2^{2d-2} + d \cdot 2^{d-1}}{n-1} \\ &= 1 - \frac{d \cdot 2^{2d-2} + d \cdot 2^{d-1}}{2^D - 1} \end{aligned}$$

■

Lemma 3:

$$\begin{aligned} &p(|K_i| = 0\ (0 \leq i < d) \setminus |C_d| = 2^d) \\ &\geq 1 - \frac{m^2}{2^D - 2^{d-1} + 1} \end{aligned}$$

Proof:

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

■

By putting them together, here comes the lower bound

$$\begin{aligned} H(X|Y_d) &\geq p(Y_d = \alpha) \cdot H(X|Y_d = \alpha) \\ &= p(|C_d| = 2^d) \\ &\quad \cdot p(|K_i| = 0 \ (0 \leq i < d) \setminus |C_d| = 2^d) \\ &\quad \cdot H(X|Y_d = \alpha) \\ &\geq (1 - \frac{d \cdot 2^{2d-2} + d \cdot 2^{d-1}}{2^D - 1}) \\ &\quad \cdot (1 - \frac{2^{d-1} md}{2^D - m + 1}) \cdot (1 - \frac{dm^2}{2^D - m + 1}) \cdot d \\ &= B(D, m, d) \end{aligned}$$

Note that the condition Y_d here contains all the information that attacker can have. It assumes that the attacker is computationally-unbounded and he has been using the system to enumerate (primary image, index) pairs for an infinite long time. Also note that the formula satisfies arbitrary number d. Thus, the lower bound of $H(X|Y)$ is the maximum value of that formula over all possible d. As a result, this is our final theorem:

Theorem 1: In defender system, one index's corresponding primary image's conditional entropy has a lower bound of

$$\max_{0 \leq d \leq D} \{B(D, m, d)\}$$

considering all the information that a computationally-unbounded attacker can have in an infinite long time. Here m is the maximum number of indexes that are allowed to be calculated between defend operations and $D = \log(n)$ denotes the logarithm of the size of primary image set.

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

$$\begin{aligned} H(X|Y_d) &= B(D, m, d) \\ &= \left(1 - \frac{c_1 \cdot D}{2^{D - 2c_1 D + 2}} + o(1)\right) \\ &\quad \cdot \left(1 - \frac{c_1 D}{2^{D - c_1 D + 1 - c_2 D}} + o(1)\right) \\ &\quad \cdot \left(1 - \frac{c_1 D}{2^{D - 2c_2 D}} + o(1)\right) \cdot c_1 D \end{aligned}$$

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

$$\begin{aligned} m &= 2^{c_2 D} = 2^{\Omega(D)} \\ H(X|Y_d) &= c_1 D + o(D) = \Omega(D) \end{aligned}$$

So following theorem is derived:

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

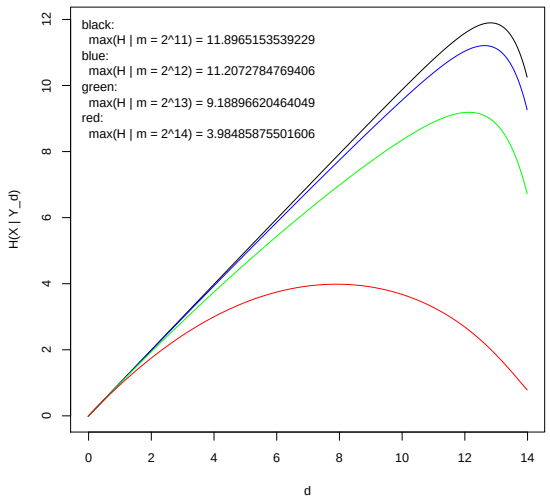

Figure 1. Lower Bound over d

C. Concrete Lower Bound and Approximate Estimation

In LiveS Cube system, $D = 32$ and m should be choosed in balance of security and efficiency.

In the following graph, the simple lower bound we proved 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 d, the x-axis is d and the peak of each line is the lower bound for each m. As it shows, when $m = 2^{12} = 4096$, the lower bound is about 11.2. This means that the system can achieve a relative high lower bound and keep its efficiency at the same time: there is only one update operation after thousands of index calculations.

In fact, the proved lower bound in theorem 1 is so simple and the tight lower bound is expected to be much higher. Observing the proof of theorem 1, only the entropy in the situation $Y = \alpha$ is count and all other entropy is considered to be 0. However, in many situations that $Y = \beta$, there is still a high entropy. What's more, $B = false$ is also assumed and the attacker is given an extra information about whether all his enumerated x in recent d updates are in candidate set C_d or not, though in real situation this is unknown to the attacker.

To have a strictly proved and very tight lower bound is a little hard. So there's an approximate lower bound which is not strictly proved, but should be more tight. In this approching, the size of candidate set is considered to be

growing in an equivalent constant rate $r \in (1, 2)$ which should be very close to 2. And in collision set K_i, each element has an equal chance of r^{-i} to become h^*. Suppose that the attacker has enumerated $f_i(x) = h$ for k rounds, i.e. $0 \leq i < k \leq \log_r(n)$. Here $i < \log_r(n)$ because when $|C_i| = n$, the information that attacker can still acquire through enumeration is considered to be 0. Under these unproved assumptions, a formal proof can be given to show that the lower bound $H(X|Y) \geq 18$ holds when $m = 2^{20}$. The detailed derivation is however too complex to write here.

D. Experimental Evaluation

For convenience, define

$$p_1 = p(B = false)p_2 \qquad = p(|C_d| = 2^d)$$
$$p_3 = p(|K_i| = 0\ (i \leq 0 < d)\ \backslash |C_d| = 2^d)$$

In our simple proof of the lower bound, p_1, p_2 and p_3 are three key points to the final result. They represent that candidate sets are maximized in last d updates, collision sets are minimized in last d updates and candidate set C_d is totally unknown to the attacker respectively. Lower bound for each of them has been proved:

$$p_1 \geq 1 - \frac{d \cdot 2^{2d-2} + d \cdot 2^{d-1}}{2^D - 1}$$
$$p_2 \geq 1 - \frac{m^2}{2^D - dm + 1}$$
$$p_3 \geq 1 - \frac{m^2}{2^D - 2^{d-1} + 1}$$

By putting them together, lower bound $H(X|Y_d)$ is achieved:

$$H(X|Y_d) \geq p_1 \cdot p_2 \cdot p_3 \cdot d$$
$$\geq B(D, m, d)$$

$p_1 \cdot p_2 \cdot p_3$ can be measured in a real program which simulates the same behaviour as we defined in defender system. So the proof above can be checked by this experimental measurement. Moreover, this experiment will show how tight our lower bound is when $H(X|Y_d = \beta)$ and $H(A|B = true)$ are ignored. Our experiment program simply simulates the whole process of d updates for 10000 times and records the number of successful events to estimate the real possibility $p_1 \cdot p_2 \cdot p_3$.

Figure 2 shows the result when $m = 2^{13}$

It can be seen that the simple lower bound is not too far away from the experimental result when $m = 2^{13}$.

Figure 3 shows the result when $m = 2^{14}$.

It seems that when m is large, the simple lower bound estimated is much smaller than experimental result. Therefore, there is still plenty of room to improve the lower bound to make it tight, even if $H(X|Y_d = \beta)$ and $H(A|B = true)$ are ignored. Meanwhile, the lower bound that can be proved

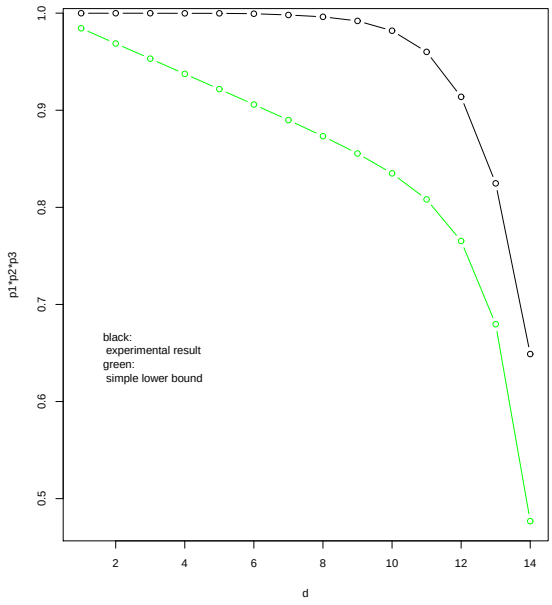

Figure 2. $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{13}$

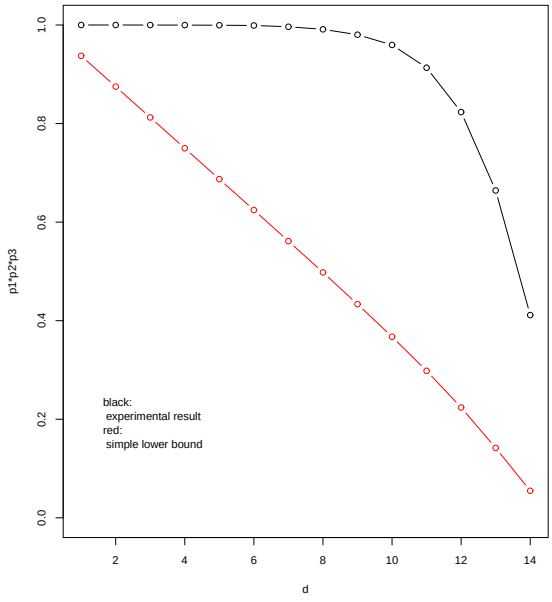

Figure 3. $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{13}$

should be higher than we simply get from theorem 1. For example, when $m = 14$, the experimental result of $p_1 \cdot p_2 \cdot p_3$ shows a lower bound of 10 when $d = 11$, while our simple lower bound only shows 4 when $d = 8$.

To check that the simple lower bound is far from the experimental result only when m is large, one more experiment

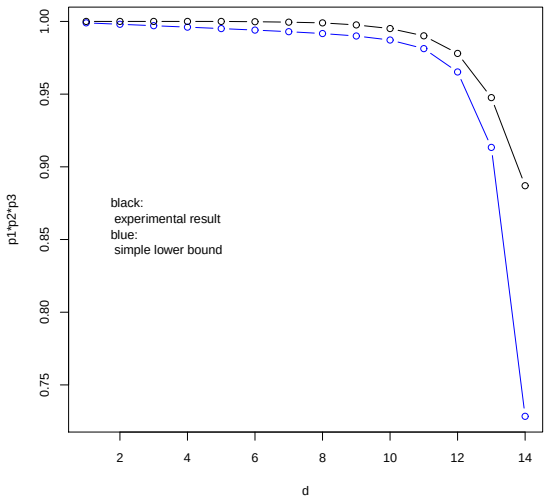

Figure 4. $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{13}$

is conducted when $m = 2^{11}$ and the result is shown in figure 4.

They are very close so now it's confirmed that our estimation of p_1, p_2, p_3 is correct and accurate when m is small.

In sum, it has been checked that our simple lower bound is valid and it is indeed not so tight even if we ignore $H(X|Y_d = \beta)$ and $H(A|B = true)$, especially when m is large. In the experiment, it shows that when $D = 32$ and $m = 2^{14} = 16384$, the lower bound is still at least 10.

E. Min-Entropy and More Meaningful Security

All the analysis above is based on Shannon entropy defined in definition 1. Min-entropy $H_\infty(X)$ define the entropy in a new way that

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

The conditional entropy could be similarly defined using min-entropy.

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

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

More specifically, $p_1 \cdot p_2 \cdot p_3$ denotes the probability that $p(x|Y = y) = 2^{-d}$. And the proof shows a lower bound of $p_1 \cdot p_2 \cdot p_3$ leading to the final lower bound of conditional Shannon entropy $p_1 \cdot p_2 \cdot p_3 \cdot d$ which is identical to conditional min-entropy. As it can be seen that min-entropy's definition is easier than Shannon entropy, it's easier to find out the practical meaning of the lower bound for conditional min-entropy. For min-entropy with a determinant condition, $H_\infty(X|Y = y) = h$ denotes the highest probability 2^{-h} that adversary can achieve to guess the right answer when $Y = y$ is known. Since $H_\infty(X|Y) = E\,(H_\infty(X|Y = y))$, conditional min-entropy just means the expected highest possibility that one adversary can guess the right answer. Therefore, the proof of our conditional entropy shows that the expected highest possibility that a computationally-unbounded adversary can compromise the security is very small: $2^{-\Omega(D)}$.

Since h is either d or 0 as in our proof, the conditional min-entropy $H_\infty(X|Y)$ is

$$\begin{aligned} &E\,(H_\infty(X|Y = y)) \\ &= p\,(H_\infty(X|Y = y) = d) \cdot d = p_1 \cdot p_2 \cdot p_3 \cdot d \end{aligned}$$

whose key is the chance to still confuse the adversary with 2^d equally possible uncertainties.

So as in the graph of $p_1 \cdot p_2 \cdot p_3$ when $m = 2^{11}$ displayed above, both simple lower bound and experimental result show that there is a chance greater than 95% percent that the adversary will be confused with 2^{12} equally possible uncertainties even if he or she is computationally-unbounded and has been attacking for an infinite long time, as long as one defend operation is enforced after 2^{11} index calculations.

[TODO floating graph]

V. Conclusion

To ensure the security of deterministic encryption for low entropy information such as cellphone numbers, a simple defender model is proposed. Though the result of this model is not strictly proved, it gives a comprehensive explanation how the security is ensured. Based on defender model, defender system is implemented in LiveS Cube system. To strictly prove its security in information theory, a simple lower bound of conditional entropy is given. Conditional entropy measures the difficulty for an adversary to get the plain information from all the related information he or she already has. The proof shows that a relative high lower bound under computationally-unbounded adversaries can be guaranteed when the efficiency is also kept. Asymptotically, suppose that the original entropy is D, a lower bound for conditional entropy of $\Omega(D)$ can be achieved when only one defend operation is required after $2^{\Omega(D)}$ attacks. And from the aspect of min-entropy, our proof shows that such an adversary is expected to compromise our security in a chance less than $2^{-\Omega(D)}$. However, the simple lower bound derived is believed to be not so tight according to an approximate lower bound and an experiment. The experiment shows that the lower bound should be much higher even if a lot of things are ignored as in the proof especially when attackers are allowed to do many attacks before one defend operation.

In sum, a strictly proved lower bound of conditional entropy in defender system is given and the tight lower bound should be still much higher than that. So it's theoretically secured and should be more secured in practical.

Though defender system is now only used in specific LiveS Cube system, it is believed that it will have a contribution to many other social network systems because deterministic encryption for low entropy information is undeniable when encrypted index or identity has to be made from information that can be easily memorized by human. This system will make it easy to eliminate security dependency of social network operators or servers and meanwhile still keep the simplicity and easy accessibility of client/server architecture than distributed architecture.

In the future work, what conditional entropy really means to real situation and what important and practical result can we derive from the lower bound of conditional entropy is worth researching. It's also worthy to establish a tighter lower bound for the system in the future, although this is a little hard.

ACKNOWLEDGMENT

The authors would like to thank... more thanks here

REFERENCES

[1] C. E. Shannon, "A mathematical theory of communication," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 5, no. 1, pp. 3–55, 2001.

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

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

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

[5] 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.

[6] 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.

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

[8] 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.

[9] 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.

[10] J. Anderson, C. Diaz, J. Bonneau, and F. Stajano, "Privacy-enabling social networking over untrusted networks," in *WOSN '09: Proceedings of the 2nd ACM workshop on Online social networks*. New York, NY, USA: ACM, 2009, pp. 1–6.

REFERENCES

[1] Theresa M. Korn; Korn, Granino Arthur. *Mathematical Handbook for Scientists and Engineers: Definitions, Theorems, and Formulas for Reference and Review*. New York: Dover Publications. pp. 613C614. ISBN 0-486-41147-8.

[2] C. Arndt (2001). *Information Measures: Information and its description in Science and Engineering)*. Berlin: Springer. pp. 370C373. ISBN 3-540-41633-1.