3-7 Error Detection and Authentication in Quantum Key Distribution

YAMAMURA Akihiro and ISHIZUKA Hirokazu

Detecting errors in a raw key and authenticating a private key are crucial for quantum key distribution schemes. Our aim is to propose practical methods for error detection and authentication in quantum key distribution schemes. We introduce several concepts about neighborhood collision free properties of Boolean functions, which are closely related to hash functions, and propose methods based on neighborhood collision free functions and error correcting codes such as Reed-Solomon code. We also examine whether or not widely used cryptographic hash functions SHA-1 and MD5 satisfy the neighborhood collision free property by computation experiments.

Keywords
Quantum cryptography, Error detection and correction, Neighborhood collision, Hash functions

1 Introduction

Quantum key distribution schemes have been introduced and studied in detail up to date (e.g.[1][2][8]). Under an ideal circumstance like an experiment in a laboratory without any physical interferences, quantum key distribution schemes enjoy the unconditional security. Since an eavesdropper Eve's unlawful access to the quantum channel causes disturbance of bit patterns of photons sent by Alice due to the Heisenberg uncertainty principle, Alice and Bob can detect Eve's intervention by estimating error rate after the data transmission through the quantum channel. Error estimation can be carried out by discussion through the classical channel. Physical errors inevitably occur in data transmission through the quantum channel under realistic circumstances. Eve may want to obtain only small amount of information concerning the private key shared by Alice and Bob. Then Eve's best strategy is to wiretap the quantum channel only small fraction of the total data transmission, and deceive Alice and Bob as if the resulting disturbance is caused by the physical defects of the quantum channel and other peripherals. By the attack, Eve may be able to obtain partial information on the private key shared by Alice and Bob. Under such a scenario, bits, where errors may have happened, are more suspicious of Eve's intervention than the other bits and should be dumped to prevent Eve from gaining any partial information. The following are essential to attain the virtually unconditional security. The first is to lower the error rate in the data transmission through the quantum channel. This depends on improvements of physical devices such as optical fibers, single photon source generators, avalanche-photo-diode detectors and so on. The error rate depends on the distance of the quantum data transmission: the longer the channel gets, the higher the error rate rises. The second is to efficiently detect (and correct) errors in the raw keys, remove the leaked information and confirm the integrity of the private key agreed by Alice

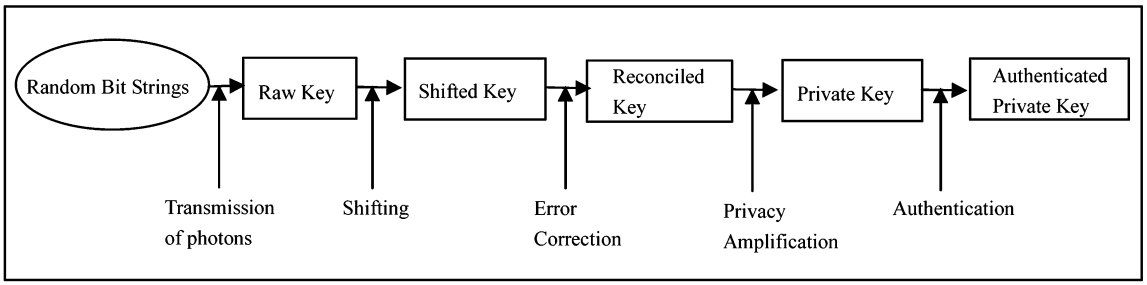


Fig.1 *Data Processing in Quantum Key Distribution*

and Bob. Our aim in this paper is to propose practical methods forward the second goal.

We briefly explain the general scheme of a quantum key distribution (see Chapter 2 of[6] for more detail). First, Alice generates a (sufficiently long) random bit string and sends photon pulses according to the random bit string through the quantum channel, where the basis and the polarization are randomly determined. Bob also generates a random bit string and measures the photon pulses with the basis determined according to his random bit string. Then Alice and Bob obtain bit strings, called raw keys, respectively. We should note that Bob's raw key is totally different from Alice' raw key because Bob does not know Alice's choice of bases and cannot get to know the bits in Alice's raw key unless he chooses the same basis. Checking their choice of bases through the classical channel, they estimate errors existing in Bob's raw key and then obtain sifted keys (this process is called sifting). The error rate is supposed to be kept under a previously fixed value, which is determined by the quality of the physical devices, unless Eve intervened. If Eve wiretapped substantial amount of data transmission from Alice to Bob through the quantum channel, Eve's intervention can be detected in this stage because Alice and Bob will find the error rate is larger than the previously fixed value. Eve's best strategy to eavesdrop is to wiretap only small fraction of the total data transmission through the quantum channel. It follows that the leaked information to Eve is at most the physical error rate.

Second, errors must be removed or corrected. After the error correction process, Alice and Bob possess an identical key called reconciled key. Note that Eve might have partial information on the reconciled key because Eve could eavesdrop the communication through the quantum and the classical channel even though the potentially leaked information is almost negligible.

Third, Eve's information is reduced substantially using privacy amplification that is the method to lower Eve's information exponentially by sacrificing bits in the reconciled key linearly ([3][4][10]). Privacy amplification can be carried out using t-resilient functions[4] (also known as (N,J,K) functions[7]). The resulting key is called a private key.

Lastly, Alice and Bob confirm the integrity of their private key and obtain an authenticated private key. We illustrate a typical process of key distribution in Fig.1 in a quantum key distribution scheme.

We introduce a concept of (a globally, locally) neighborhood collision free function} and show that SHA-1[9] and MD5[12] enjoy the neighborhood collision free property by experiments with computers. We present methods to detect errors in the raw keys and to authenticate the private key in a quantum key distribution scheme using a neighborhood collision free function. Our methods realize the error detection (correction) and authentication procedures in Fig.1.

2 Several error correction methods

We briefly explain the error correction methods in [4] and [5] in this section. Suppose Alice and Bob possess their sifted keys after the sifting process in a quantum key distribution scheme. If Alice has a sifted key r, then Bob has a sifted key $r \oplus e$, where $\oplus$ denotes the bitwise exclusive or, and e represents the errors occurred. The Hamming weight of e depends on the physical error rate of data transmission through the quantum channel, and the recent physical experiments show relatively low error rate for short distance transmission. The physical error rate is the fraction of occurrence of errors in the total data transmission through the quantum channel. Under the most ideal assumption, we have $e = 0$, and hence, Alice and Bob share the identical key, on which Eve has no chance to get any information on it. Although physical errors unavoidably occur at some rate under the realistic situation, they are very rare. Therefore, the Hamming weight of e is in proportion to the error rate and so slightly greater than 0. We may assume that most of bits in e are 0. To share the identical private key, Alice and Bob need to get rid of the error bits. Especially, if they intend to use the key as the secret key for a symmetric cipher, it is crucial to share an identical authenticated private key.

First, we explain the error correction method by Bennett, Bessette, Brassard, Salvail and Smolin [5]. Alice divides her sifted key into blocks. Bob also divides his sifted key in the same way as Alice does: if Alice has the sifted key r and r is divided as $r = r_1 r_2 \cdots r_n$, then Bob has the sifted key $r \oplus e$ and it is divided as $r \oplus e = (r_1 \oplus e_1)(r_2 \oplus e_2) \cdots (r_n \oplus e_n)$, where $e = e_1 e_2 \cdots e_n$ represents the error bits. Then Alice computes the parity of each block r_i and sends them all to Bob through the classical channel. Eve can wiretap the classical channel and is able to obtain the parities of the blocks. The parity of each block is considered as one bit information, and so, Alice and Bob take it for granted that one bit information is leaked for each block. Bob computes the parities of the corresponding blocks of his sifted key and compares them with the parities sent by Alice. If all of them coincide, then Alice and Bob probably possess the identical key. Otherwise, some of Alice's block and Bob's block must be different at least one position. In such a case, Alice and Bob divide the block whose parities are different into shorter blocks and continue the process until they do not find any different parity. In any stage, Alice and Bob delete one bit from each block at the same position in order to make the leaked information to Eve meaningless. Repeating the process several times, Alice and Bob eventually establish an identical key with a high probability. Demerits of this method are following: Alice and Bob are not guaranteed to share the identical reconciled key. It wastes numerous bits and requires considerable computation. In the process of generating raw keys, Alice and Bob cannot theoretically predict the number of necessary bits to establish the reconciled key, that is, it is quite hard to theoretically estimate the efficiency of the error correction.

Second, we explain one of the methods in Bennett, Brassard and Robert [4]. They proposed that Alice sends the hash value of her sifted key through the classical channel. Bob computes the hash value of his sifted key as well. Bob compares these two hash values. If they are identical, they share the identical reconciled key. Otherwise, Bob turns around a few bits in his sifted key, computes the hash value of the altered key then and checks whether or not it coincides with the hash value of Alice's sifted key. Bob continues this process until he finds the one whose hash value coincides with the hash value of Alice's sifted key. Bob basically carries out the exhaustive search to find positions in his bit string, where the errors happen, until he detects the errors. The method is called a bit twiddling. The defect of the method is that Bob is required to carry out substantial computation, and the hash value transmitted through the classical channel gives substantial

information to Eve as well. Only under the very restricted assumption that the error rate is very low and the bit string is short, the exhaustive search can be carried out. Otherwise, the task is impossible. It is also proposed in [4] that Alice encodes her sifted key by an error correcting code and sends only the redundancy part of the encoded sifted key. The defect of this method is again that the redundancy part of encoded sifted key gives substantial information to Eve. This method has several demerits, nevertheless, these can be remedied as we will see in Section **4**.

3 Neighborhood collision free functions

Let H be a Boolean function of Z_2^l to Z_2^k. Intuitively, H is neighborhood collision free if H maps any two bit strings with a small Hamming distance to bit strings with a large Hamming distance. Recall that the Hamming distance of bit strings x_1 and x_2 is the number of positions where the entry of x_1 is different from that of x_2. The Hamming weight of a bit string x is the Hamming distance between x and the zero (that is, the string consisting of only 0). This property should be satisfied by all (symmetric and asymmetric) encryption functions, although it is not sufficient for secure communication. Recall that a Boolean (hash) function H is (strongly) collision free if it is hard to find bit strings r_1 and r_2 with $r_1 \neq r_2$ and $H(r_1) = H(r_2)$. In other words, H is (strongly) collision free if it is hard to find bit strings $r_1$and r_2 such that $r_1 \neq r_2$ and the Hamming distance between $H(r_1)$ and $H(r_2)$ is 0. This concept is generalized as follows. Let us denote the Hamming distance between r and s by d(r, s), where $r, s \in Z_2^l$. For $t \in Z_2^l$, the set $\{s \in Z_2^l | d(s, t) < i\}$ is called the neighborhood around t of radius i and denoted by N(t, i). We define several neighborhood collision free properties. Let H be a Boolean function of Z_2^l to Z_2^k.

- H is a globally j-neighborhood collision free function if it is hard to find s, $t \in Z_2$ such that $H(s) \in N(H(t), j)$, equivalently $H(t) \in N(H(s), j)$ (or $N(\{H(s), j/2\} \cap N(H(t), j/2))$ is not empty).
- H is a locally j-neighborhood collision free function in i-neighborhood if for every $u \in Z_2^l$ it is hard to find $s,t \in N(u,i)$ such that $H(s) \in N(H(t), j)$, equivalently $H(t) \in N(H(s), j)$ (or $N(H(s), j/2) \cap N(H(t), j/2$ is not empty).
- H is a globally collision free function if it is hard to find s, $t \in Z_2^l$ such that $H(s) = H(t)$.
- H is a locally collision free function in i-neighborhood if for every $u \in Z_2^l$ it is hard to find s, $t \in N(u,i)$ such that $H(s) = H(t)$.

These concepts play a vital role in construction of our error detection and authentication scheme. The concept of the hardness depends on the context, and it may be information theoretic or computational. A globally collision free property coincides with a (strongly) collision free property for cryptographic hash functions. It is easy to see that a globally j-neighborhood collision free function is a locally j-neighborhood collision free function in i-neighborhood, a globally j-neighborhood collision free function is a globally collision free function, a globally collision free function is a locally collision free function in j-neighborhood and a locally j-neighborhood collision free function in i-neighborhood is a locally collision free function in i-neighborhood. The converses are not necessarily true. See Fig.2 for the relationships among the concepts.

For example, good block ciphers show the strong avalanche effect, and hence, they satisfy the globally neighborhood collision free property even under a low round. The globally neighborhood collision free property can be considered as a generalization of the avalanche effect. We shall show, in Section **5**, that SHA-1 and MD5 satisfy the globally neighborhood collision free property by experiments by computers. Our experiments show that SHA-1 has the 43-neighborhood collision free property, and MD5 has the 34-neighborhood collision free property, however, it is dif-

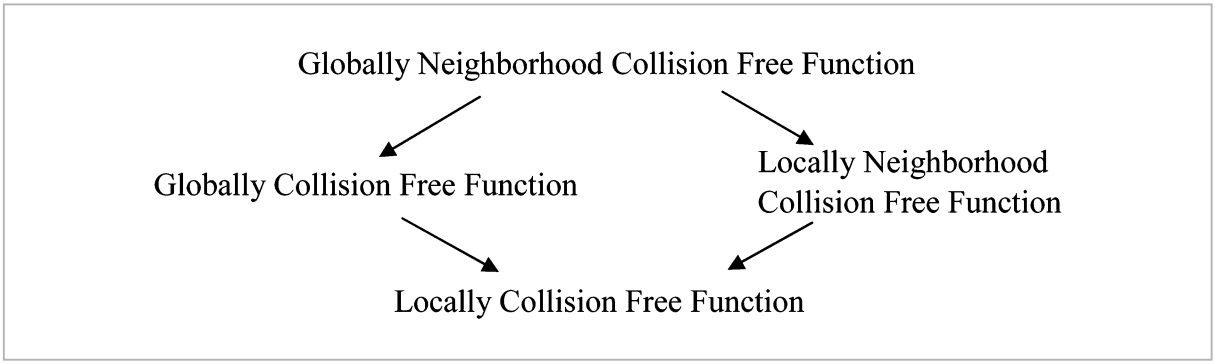


Fig.2 *Hierarchy of Collision Free Functions*

ficult to prove theoretically and rigorously that they really do.

4 Error detection using locally neighborhood collision free functions

The methods explained in Section **2** waste numerous bits and require considerable computation such as iterations of random permutations to detect and correct errors. Moreover, it is difficult for us to predict the number of necessary bits, that is, the length of raw keys, to succeed in establishing an authenticated private key in the final stage. It is desired to invent a simple efficient method so that we can predict easily and theoretically the number of necessary bits in advance. We employ a locally neighborhood collision free function to detect errors in the sifted keys.

Suppose that the physical error rate of the quantum data transmission is $\varepsilon>0$. We note that Alice and Bob should operate a random permutation to their sifted keys after the sifting process. If they have done so, we can suppose the errors are random, that is, the errors are uniformly distributed in Bob's sifted key. If Eve eavesdrops the bits located at specific positions in the private key (according to his eavesdropping strategy) and Alice and Bob do not operate a random permutation, then the errors are burst, that is, they are distributed non-uniformly in Bob's sifted key. After the error estimation process, Alice and Bob have their sifted keys, r and s, where $r, s \in Z_2^l$.Then $r \oplus s$ shows the error bit pattern and its Hamming weight is approximately $\varepsilon \times l$. Suppose $0<\varepsilon<1$ and $0<\alpha<1$ are constants such that α is sufficiently larger than ε. Let H be a locally neighborhood collision free function of Z_2^l to Z_2^k with $\theta(H, \varepsilon, \alpha)$ that is the probability of the event $d(H(r_1), H(r_{12})) \leqq \alpha \times k$ when we choose randomly and uniformly a pair (r_1, r_2) of distinct bit strings from Z_2^l such that the Hamming distance between r_1 and r_2 is less than or equal to $\varepsilon \times l$. A Boolean function H is considered locally neighborhood collision free if $\theta(H, \varepsilon, \alpha)$ is negligible for some constants ε and α such that $0<\varepsilon \ll \alpha<1$.

We now explain the basic idea of an error detection method. Suppose that H is a locally neighborhood collision free function and $\theta = \theta(H, \varepsilon, \alpha)$ is small. This implies that the probability that $d(H(r),H(s))<\alpha \times k$ for $r \neq s \in Z_2^l$ with $d(r, s)<\varepsilon \times l$ is negligible. We assume the Hamming weight of $r \oplus s$ is less than $\varepsilon \times l$. Hence, if $r \neq s$, then H(s) is not in $N(H(r), \alpha \times k)$, equivalently the Hamming weight of $H(r) \oplus H(s)$ is bigger than $\alpha \times k$, by the locally neighborhood collision free property of H. If r = s, H(r) = H(s) and so the Hamming weight of $H(r) \oplus H(s)$ is 0.

We now suppose Alice and Bob possess t and $t \oplus e$ as parts of their sifted keys, respectively, where $t, e \in Z_2^k$ and the Hamming weight of e is approximately $\varepsilon \times k$. Then the Hamming distance between $H(r) \oplus t$ and $H(s) \oplus (t \oplus e)$ is given by $(\{H(r) \oplus t\} \oplus (H(s) \oplus (t \oplus e)) = (H(r) \oplus H(s)) \oplus e$. Hence, the Hamming distance is approximately $\varepsilon \times k$ if r = s, otherwise, it is more than $\alpha \times k$. So if we set $(\varepsilon+\alpha)k/2$ as a threshold, Bob can determine

whether or not r = s by checking whether the Hamming distance between $H(r) \oplus t$ and $H(s) \oplus (t \oplus e)$ is smaller or bigger than $(\varepsilon + \alpha)k/2$.

We combine this criterion to find the existence of errors and several methods to find the exact bit positions where the errors occurred. We discuss several methods in the following subsections. The difference among the first three methods lies in the consumption of resources (computation, quantum data transmission and classical data transmission). This difference indicates the existence of a trade-off relation among computation, quantum communication and classical communication.

4.1 Method

Suppose that l is the intended size of a reconciled key. Let H be a locally neighborhood collision free function of Z_2^l to Z_2^k such that the probability $\theta(H, \varepsilon, \alpha)$ is negligible and $\varepsilon \ll \alpha$. We assume Alice and Bob can make use of H. Note that H is not necessarily kept secret, and hence, Eve can also make use of it. Alice and Bob first establish 2l + k bit sifted keys in the sifting process. Alice and Bob have 2l + k bit binary strings r and $r \oplus e$ as their sifted keys, respectively. Here, erepresents the errors. The Hamming weight of e is approximately $\varepsilon \times |e| = \varepsilon \times l$. The basic idea is that Alice and Bob sacrifice l+k bits of their sifted keys and detect error bits in e without leaking any information to Eve. Then they share r and agree that r is their reconciled key.

Suppose Alice has r as her sifted key and $r = r_1 r_2 r_3$, where $r_1, r_2 \in Z_2^l$ and $r_3 \in Z_2^k$ r_1, r_2. Alice computes the hash value $H(r_1)$, then sends $r_1 \oplus r_2$ and $H(r_1) \oplus r_3$ to Bob through the classical channel. Eve can wiretap the classical channel. Bob has $r \oplus e$ as his sifted key and $r \oplus e = (r_1 \oplus e_1)(r_2 \oplus e_2)(r_3 \oplus e_3)$, where $e = e_1 e_2 e_3$ and $e_1, e_2 \in Z_2^l$ and $e_3 \in Z_2^k$. Bob, receives $r_1 \oplus r_2$ and $H(r_1) \oplus r_3$. Thus, Bob possesses $r_1 \oplus e_1$, $r_2 \oplus e_2$, $r_3 \oplus e_3$, $r_1 \oplus r_2$, $H(r_1) \oplus r_3$. He computes the hash value $H(r_1 \oplus e_1)$. Next he computes $(r_1 \oplus r_2) \oplus (r_2 \oplus e_2) = r_1 \oplus e_2$ and $(r_1 \oplus e_2) \oplus (r_1 \oplus e_1) = e_1 \oplus e_2$. The bit string $e_1 \oplus e_2$ contains considerable information on the bit string $e_1 e_2$. Bob now computes $(H(r_1) \oplus r_3) \oplus (r_3 \oplus e_3) = H(r_1) \oplus e_3$ and $(H(r_1) \oplus e_3) \oplus (H(r_1) \oplus e_1) = H(r_1) \oplus (H(r_1) \oplus e_1) \oplus e_3$. If e_1 contains no 1, that is, $r_1 = r_1 \oplus e_1$, then we have $H(r_1) = H(r_1 \oplus e_1)$. In this case, $H(r_1) \oplus (H(r_1) \oplus e_1) \oplus e_3 = e_3$. Hence, the Hamming weight of $H(r_1) \oplus (H(r_1) \oplus e_1) \oplus e_3$ is smaller than $(\alpha + \varepsilon)k/2$ with a high probability. On the other hand, if e_1 contains 1, then $H(r_1) \oplus (H(r_1) \oplus e_1) \oplus e_3$ is larger than $(\alpha + \varepsilon)k/2$ with a high probability. So we can decide whether or not $e_1 = 0$ by the threshold criterion that Hamming weight of $H(r_1) \oplus (H(r_1) \oplus e_1) \oplus e_3$ is bigger than or smaller than $(\alpha + \varepsilon)k/2$.

If e = 0, then Alice and Bob established the identical key r_1 of size l. If $H(r_1) \neq H(r_1 \oplus e_1)$, then Bob guesses e_1 from the information $e_1 \oplus e_2$ (bit twiddling). Then he computes the hash values of the bit string twiddled from $r_1 \oplus e_1$ according to the information $e_1 \oplus e_2$ and compares them with $H(r_1) \oplus e_3$. Bob can eventually finds e' such that $H(r_1) = H(r_1 \oplus e_1 \oplus e')$ (strictly speaking, e' such that the Hamming distance between $H(r_1) \oplus e_3$ and $H(r_1 \oplus e_1 \oplus e')$ is smaller than $(\alpha + \varepsilon)k/2$. Since H is locally neighborhood collision free, it is implausible that he finds $e' \neq e_1$ and $H(r_1) = H(r_1 \oplus e_1 \oplus e')$. Hence, $e' = e_1$ holds with a high probability and Bob can detect all errors occurred in quantum data transmission. Alice and Bob can delete or correct these error bits $e_1 = e'$ and establish a reconciled key r_1' of the length slightly shorter than l (when the errors are deleted). We should note that if Alice and Bob correct (not to delete) and reuse the error bits, then they share the reconciled key r_1' of exactly size l. Amplifying privacy, they can reduce enemy's information at their own will.

We briefly discuss the security of the method. Eve can only obtain information out of communication through the classical channel under the assumption that the process of establishing the shifted key is sound. Thus, Eve can obtain only $r_2 \oplus r_2$ and $H(r_1) \oplus r_3$. By the mechanism of quantum key distribution scheme, r_1, r_2, r_3 are mutually independent random bit strings. We can consider r_1 and

$H(r_1)$ are encrypted by the one-time pad, also known as the Vernam encryption[14], sacrificing r_2 and r_3, respectively. This implies that Eve can obtain virtually no information as the one-time pad enjoys the perfect secrecy [13]. However, physical implementation problem leaves room for Eve to obtain small amount of information. In the case that Eve wiretapped only small fraction of the total data transmission, succeeded in her attack and obtained partial information of the reconciled key r_1, the information is estimated at most $2\varepsilon \times l$ bits. This leaked information can be removed by the privacy amplification process.

4.2 Method 2

We suppose Bob has strong computation power and then discuss a method to reduce the amount of quantum data transmission by demanding Bob substantial computation as a trade-off. Data transmission through the quantum channel costs much more than data transmission through the classical channel and computation, and hence, it is reasonable to require Bob to perform substantial computation if he has abundant computation resource. As before, H is a locally neighborhood collision free function of Z_2^l to Z_2^k such that the probability $\theta(H, \varepsilon, \alpha)$ is negligible and $\varepsilon \ll \alpha$.

Suppose that Alice has r_1 r_2 as her sifted key, where $r_1 \in Z_2^l$ and $r_2 \in Z_2^k$ r_1, whereas Bob has $(r_1 \oplus e_1)(r_2 \oplus e_2)$ as his sifted key, where e_1 and e_2 represent the errors. Alice computes the hash value $H(r_1)$ and sends $H(r_1) \oplus r_2$ to Bob through the classical channel. The communication can be considered encrypted by the one-time pad. Note that the amount of bits transmitted is the constant k. Bob computes $H(r_1 \oplus e_1)$ and $(H(r_1) \oplus r_2) \oplus (r_2 \oplus e_2) = H(r_1) \oplus e_2$ and its Hamming weight is approximately $k \times \varepsilon$. If $H(r_1) \neq H(r_1 \oplus e_1)$, then the Hamming weight of $(r_1 \oplus e_1) \oplus H(r_1) \oplus e_3$ is approximately $k \times \alpha$ since H is locally neighborhood collision free. Since α is sufficiently larger than ε, we can conclude with a high probability that $H(r_1) = H(r_1 \oplus e_1)$ if the Hamming weight of $H(r_1) \oplus e_2$ is smaller than $(\alpha + \varepsilon)k/2$, and $H(r_1) \neq H(r_1 \oplus e_1)$ otherwise. If $H(r_1) \neq H(r_1 \oplus e_1)$, then Bob twiddles randomly up to $\varepsilon \times l$ bits of $r_1 \oplus e_1$, computes the hash values of them and then compares with $H(r_1) \oplus r_2$. Bob can eventually find e_1 by the exhaustive search, however, e_1 has approximately $\varepsilon \times l$ bits of 1 and so Bob twiddles only up to about $\varepsilon \times l$ bits of $r_1 \oplus e_1$. Clearly Bob's computation task depends on the length of r_1 and the error rate ε.

Let us discuss the amount of data transmission through the quantum and classical channels. In Method 1, Alice and Bob have to generate sifted keys of size 2l+k to generate a reconciled key of length l bits. The amount of the quantum data transmission is proportion to 2l+k. The amount of the classical data transmission is l+k. In Method 2, on the other hand, the amount of the quantum data transmission is proportion to l+k and the amount of the classical data transmission is k.

Another merit in Method 2 is that information potentially leaked to Eve is reduced compared with Method 1. The reason is that the total (quantum and classical) communication is less than in Method 1. In Method 1, it is estimated that Eve may have stolen at most $\varepsilon \times (2k+l)$, whereas in Method 2, at most $\varepsilon \times (k+l)$.

A defect of Method 2 is to require Bob considerable amount of computation. If ε is small and the length of the established key is small, then Bob's computation can be carried out by a desktop computer. However, if ε is large and the key length is long, then the computation becomes an impossible task.

4.3 Method 3

We give an intermediate between Method 1 and Method 2. Suppose H is a locally neighborhood collision free function of Z_2^l to Z_2^k such that the probability $\theta(H, \varepsilon, \alpha)$ is negligible and $\varepsilon \ll \alpha$. Alice has r_1 r_2 r_3 r_4 as her sifted key and r_1, r_2, $r_3 \in Z_2^{l/2}$ and $r_4 \in Z_2^k$ r_1, r_2, r_3. Similarly Bob has $(r_1 \oplus e_1)(r_2 \oplus e_2)(r_3 \oplus e_3)$ $(r_4 \oplus e_4)$ as his sifted key, where e_1, e_2, $e_3 \in Z_2^{l/2}$ and $e_4 \in Z_2^k$. The string e_1 e_2 e_3 e_4 represents the errors. Alice and Bob intend to establish a reconciled key r_1 r_2. The bit string e_1 e_2 contains approximately $\varepsilon \times l$ bits of 1.

Alice computes $r_1 \oplus r_2 \oplus r_3$ and $H(r_1\ r_2) \oplus r_4$ and sends it to Bob through the classical channel. Bob computes $(r_1 \oplus r_2 \oplus r_3) \oplus (r_3 \oplus e_3) = r_1 \oplus r_2 \oplus e_3$ and $(H(r_1\ r_2) \oplus r_4) \oplus (r_4 \oplus e_4) = H(r_1\ r_2) \oplus e_4$. He computes $(r_1 \oplus e_1) \oplus (r_2 \oplus e_2) = r_1 \oplus r_2 \oplus (e_1 \oplus e_2)$, and then $(r_1 \oplus r_2 \oplus e_3) \oplus (r_1 \oplus r_2 \oplus (e_1 \oplus e_2)) = e_1 \oplus e_2 \oplus e_3$. If $r_1\ r_2$ is equal to $(r_1 \oplus e_1)(r_2 \oplus e_2) = (r_1\ r_2) \oplus (e_1\ e_2)$, then the Hamming distance between $H(r_1\ r_2) \oplus e_4$ and $H((r_1 \oplus e_1)(r_2 \oplus e_2))$ is approximately $\varepsilon \times k$. On the other hand, if $r_1\ r_2$ is not equal to $(r_1 \oplus e_1)(r_2 \oplus e_2))$, then the Hamming distance between $H(r_1\ r_2) \oplus e_4$ and $H((r_1 \oplus e_1)(r_2 \oplus e_2))$ is more than $\alpha \times k$. Since α is sufficiently larger than ε, Bob can decide whether or not $e_1\ e_2 = 0$ by the threshold criterion that the Hamming distance between $H(r_1\ r_2) \oplus e_4$ and $H((r_1 \oplus e_1)(r_2 \oplus e_2))$ is bigger or smaller than $(\varepsilon+\alpha)k/2$. If $H(r_1\ r_2)=H((r_1 \oplus e_1)(r_2 \oplus e_2))$, then Alice and Bob agree the reconciled key $r_1\ r_2$. If $H(r_1\ r_2) \neq H((r_1 \oplus e_1)(r_2 \oplus e_2))$, then Bob guesses $e_1\ e_2$ using the information $e_1 \oplus e_2 \oplus e_3$ (bit twiddling). Clearly it is much easier to find $e_1\ e_2$ than Method 2, but more difficult than Method 1.

For Alice and Bob to establish a reconciled key of length l, $r_1\ r_2$ must be of length l. Note that $|r_1|=|r_2|=|r_3|= 1/2$ and $|r_4|= k$. Hence, Alice and Bob have to generate a sifted key of length $3l/2+k$. If we ignore k, they need to generate a bit string of length almost $3l/2$ of the reconciled key length whereas sifted keys of size 2l and l are required in Method 1 and Method 2, respectively.

4.4 Method using error correcting codes

We briefly discuss a method using error correcting codes. Suppose H is a locally neighborhood collision free function of Z_2^l to Z_2^k such that the probability $\theta(H, \varepsilon, \alpha)$ is negligible and $\varepsilon \ll \alpha$. To correct the errors in sifted keys of Alice and Bob, Alice may want to encode her sifted key by a classical error correcting code and transmit only the redundancy part of the encoded sifted key. However, the redundancy part gives substantial information of Alice's sifted key, and hence, the redundancy part must be encrypted to prevent Eve from obtaining any information. We propose to encrypt the redundancy part by the one-time pad. Suppose Alice has $r_1\ r_3$ as her sifted key, where $r_1 \in Z_2^l$ and $r_3 \in Z_2^k$, and Bob has $(r_1 \oplus e_1)(r_3 \oplus e_3)$ as his sifted key, where $e_1 \in Z_2^l$ and $e_3 \in Z_2^k$. Alice computes the redundancy (denoted by $C(r_1)$) of the encoded word of r_1 by the error correcting code C. Bob can detect and correct the error bit string e_1 if he has most correct bits of $C(r_1)$ with his sifted key $r_1 \oplus e_1$. Alice sends $C(r_1) \oplus e_3$, and hence, $C(r_1)$ is encrypted by the one-time pad and so it gives virtually no information to Eve even if she can eavesdrop it. Bob can compute $C(r_1) \oplus e_3 \oplus (r_3 \oplus e_3) = C(r_1) \oplus r_3$. Hence, if the error rate is small enough, then Bob can correct the error bits due to the error-correcting ability of C. For instance, we can use the Reed-Solomon code[11] for our purpose because of its capability of correcting random errors. Note that we may assume that errors distribute uniformly all over the sifted keys because Alice and Bob operated a random permutation to their sifted keys after the sifting process.

4.5 Authentication

After generating a reconciled key, Alice and Bob carry out privacy amplification and obtain their private key. Next they confirm the integrity of their private key. We can employ the same idea to authenticate a private key. We should note that the existing methods basically require the previously shared authenticated private key, while ours do not. Suppose that after the privacy amplification process, Alice has her private key r_1 and Bob has his private key r_1', where $r_1, r_1' \in Z_2^l$. When making their raw keys, Alice and Bob generate extra sifted keys r_3 and $r_3 \oplus e_3$, respectively, where $r_3 \in Z_2^k$ and e_3 represents the errors. Alice sends $H(r_1) \oplus r_3$ to Bob. This transmission is considered as encrypted by the one-time pad, and hence, Eve obtains virtually no information. Bob checks whether or not the Hamming distance between $H(r_1) \oplus r_3$ and $H(r_1 \oplus e_1)$ is smaller than the threshold $(\varepsilon+\alpha)k/2$. If so, r_1

Table 3 *Static of Experiments on SHA-1 and MD5*

Algorithm	ID	#data	MEAN	S.D	Max.h.d	Min.h.d
SHA-1	1	10^8	80.000029	6.327076	115	44
	10	10^8	80.004204	6.334314	109	49
	20	10^8	79.994482	6.321717	111	47
MD5	1	10^8	63.999359	5.656389	95	34
	10	10^8	63.998326	5.658194	93	38
	20	10^8	63.995178	5.655455	92	37

$= r_1 \oplus e_1$ and $e_1 = 0$, $r_1 \neq r_1 \oplus e_1$. This authentication method can be applied after the error correction process. We also note that the method can be employed after any error correction and privacy amplification method.

4.6 Experimental results

To implement our error detection method, we need a concrete locally neighborhood collision free function. We show by experiment with computers that SHA-1[9] and MD5[12] satisfy the locally neighborhood collision free property. If a function H satisfies the locally neighborhood collision free property, then the Hamming distance of $H(x_1)$ and $H(x_2)$ is expected to be relatively large with a high probability for any bit strings x_1, x_2 having a small Hamming distance. In our experiments, we choose randomly N = 100,000,000 pairs (x_1, x_2) of bit strings having Hamming distance 1 (10, 20, respectively). Then we count the frequency of the Hamming distance of the pair $(H(x_1), H(x_2))$. If H is a cryptographic hash function, we easily imagine that H exhibits a normal distribution. If the standard deviation is relatively small, that is, most samples yields a Hamming distance close to the mean value, then we can conclude that it is a good neighborhood collision free function.

We consider SHA-1 as a function of Z_2^{512} into Z_2^{160}, that is, we restrict its domain to Z_2^{512} in our experiments. We expect the mean value to be 80, and Hamming distance $d(H(x_1), H(x_2))$ is close to 80 for most pairs (x_1, x_2). Actually, our experiments for SHA-1 with 10,000,000 samples of Hamming distance 1 (10, 20) show that the mean value is about 80, the standard deviation is 6.3, the minimum of $d(H(x_1), H(x_2))$ is 44, and the maximum of $d(H(x_1), H(x_2))$ is 115. See Table 1 for the statistic and Fig.4 and Fig. 5 for the histograms in Appendix. Our experiments show that the deviation is small enough. Hence, SHA-1 has the good neighborhood collision free property, and hence, most pairs of bit strings with Hamming distance 1 are mapped to the strings with Hamming distance close to 80. For example, we may set $\alpha = 1/4$. Then the probability $\theta(H, \varepsilon, \alpha)$ is negligible for any error rate $0 < \varepsilon < \alpha$. In this case, the threshold value is around $(\varepsilon + (1/4)) \times 180)/2$.

We consider MD5 as a function of Z_2^{512} to Z_2^{128}. Hence, we expect the mean value to be 64, and Hamming distance $d(H(x_1), H(x_2))$ is close to 64 for most pairs (x_1, x_2). Our experiments for MD5 with 10,000,000 samples of Hamming distance 1 (10, 20) show that the mean value is about 64, the standard deviation is 5.6, the minimum of $d(H(x_1), H(x_2))$ is 34, and the maximum of $d(H(x_1), H(x_2))$ is 95. See Table 3 for the statistic and Fig. 4 and Fig. 5 for the histograms in Appendix. Our experiments show that the deviation is small enough. Hence, MD5 has the good neighborhood collision free property, and hence, most pairs of bit strings with Hamming distance 1 are mapped

Table 4 Hamming distance histogram of SHA-1

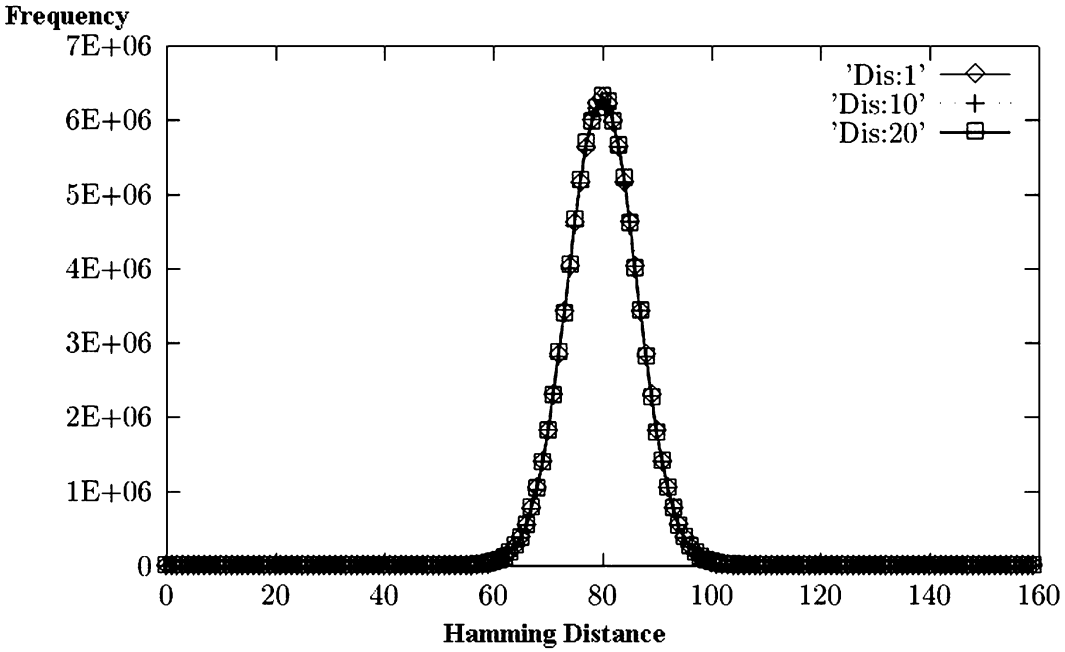
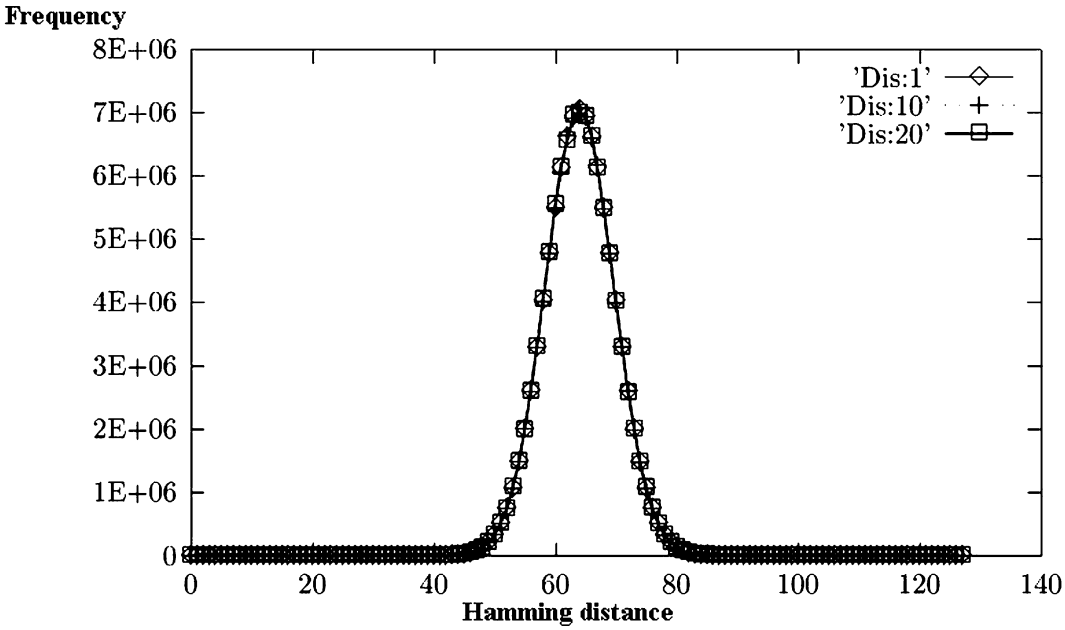


Table 5 Hamming distance histogram of MD5



to the strings with Hamming distance close to 64. For example, we may set $\alpha = 1/4$. Then the probability $\theta(H, \varepsilon, \alpha)$ is negligible for any error rate $0<\varepsilon<\alpha$. In this case, the threshold value is around $(\varepsilon+(1/4))\times 128)/2$.

In Table 4 and Table 5, the graph labeled Dis:1, Dis:10, Dis:20 indicates the histogram of Hamming distance 1, 10, 20.

References

1 C.H.Bennett and G.Brassard, "Quantum cryptography : Public-key distribution and coin tossing", Proc. Int. Conf. on Computers, Systems and Signal Processing, Bangalore, India, pp.175-179, 1984.

2 C.H.Bennett, "Quantum Cryptography Using Any Two Nonorthogonal States", Phys. Rev. Lett., Vol.68, pp.3121-3124, 1992.

3 C.H.Bennett, G.Brassard, C.Crepeau, and U.M.Maurer, "Generalized privacy amplification", IEEE Trans. Information Theory, Vol.41, pp.1915-1923, 1995.

4 C.H.Bennett, G.Brassard, and J.M.Robert, "Privacy amplification by Public Discussion", SIAM J Comput., Vol.17, pp.210-229, 1988.

5 C.H.Bennett, F.Bessette, G.Brassard, L.Salvail, and J.Smolin, "Experimental Quantum Cryptography", J.Cryptology, Vol.5, pp.3-28, 1992.

6 D.Bouwmeester, A.Ekert, and A.Zeilinger, "The Physics of Quantum Information", Springer- Verlag, Berlin Heidelberg New York, 2000.

7 B.Chor, O.Goldreich, J.Hastad, J.Freidmann, S.Rudich, and R.Smolensky, "The Bit Extraction Problem or t-resilient Functions", 26th IEEE Symp. Foundations of Computer Science, pp.396- 407, 1985.

8 A.K.Ekert, "Quantum Cryptography Based on Bell's Theorem", Phys. Rev. Lett. Vol.67, No.6, pp.661-663, 1991.

9 FIPS 180-1 : Secure Hash Standard, Federal Information Processing Standard (FIPS), Publication 180-1, National Institute of Standards and Technology, US Department of Commerce, Washington D.C., April, 1995.

10 U.M.Maurer, "Secret Key Agreement by Public Discussion from Common Information", IEEE Trans. Information Theory, Vol.39, pp.733-742, 1993.

11 I.S.Reed and G.Solomon, "Polynomial Codes over Certain Finite Fields", J.Soc. Indust. Appl. Math. Vol.8, pp.300-304, 1960.

12 R.L.Rivest, "The MD5 Message-digest algorithm", Request for Comments (RFC) 1321, Internet Activities Board, Internet Task Force, April, 1992.

13 C.E.Shannon, "Communication Theory of Secrecy Systems", Bell Syst. Tech. J., Vol.28, pp.656- 715, 1948.

14 G.S.Vernam, "Cipher Printing Telegraph Systems for Secret Wire and Radio Telegraphic Communications", J.Amer. Inst. Elect. Eng., Vol.55, pp.109-115, 1926.

15 H.Zbinden, H.Bechmann-Pasquinucci, N.Gisin, and G.Ribordy, "Quantum Cryptography", Applied Physics B, Vol.67, pp.743-748, 1998.

16 A.Yamamura and H.Ishizuka, "Detecting errors and authentication in quantum key distribution", Information Security and Privacy (ACISP2001), LNCS 2119, Springer-Verlag, pp.260-273, 2001.

***YAMAMURA Akihiro**, Ph.D.*

Group Leader, Security Fundamentals Group, Information and Networks Systems Department

Information security, Cryptography, Algebraic systems and their algorithms

ISHIZUKA Hirokazu

Head Researcher, Information Technology R&D Center, MITSUBISHI ELECTRIC CORPORATION

Quantum Cryptography, Quantum Information Technology