IMES DISCUSSION PAPER SERIES

Year 2010 issues on cryptographic algorithms

Masashi Une and Masayuki Kanda

Discussion Paper No. 2006-E-8

IMES

INSTITUTE FOR MONETARY AND ECONOMIC STUDIES

BANK OF JAPAN

C.P.O BOX 203 TOKYO
100-8630 JAPAN

You can download this and other papers at the IMES Web site:
http://www.imes.boj.or.jp

IMES Discussion Paper Series 2006-E-8
June 2006

Year 2010 issues on cryptographic algorithms

Masashi Une[†] and Masayuki Kanda[*]

Abstract

In the financial sector, cryptographic algorithms are used as fundamental techniques for assuring confidentiality and integrity of data used in financial transactions and for authenticating entities involved in the transactions. Currently, the most widely used algorithms appear to be two-key triple DES and RC4 for symmetric ciphers, RSA with a 1024-bit key for an asymmetric cipher and a digital signature, and SHA-1 for a hash function according to international standards and guidelines related to the financial transactions.

However, according to academic papers and reports regarding the security evaluation for such algorithms, it is difficult to ensure enough security by using the algorithms for a long time period such as ten or fifteen years due to advances in cryptanalysis techniques, improvement of computing power and so on. In order to enhance the transition to more secure ones, NIST (National Institute of Standards and Technology) of the United States describes in various guidelines that NIST will no longer approve two-key triple DES, RSA with a 1024-bit key, and SHA-1 as the algorithms suitable for IT systems of the Federal Government after 2010.

It is an important issue how to advance the transition of the algorithms in the financial sector. This paper refers to issues regarding the transition as Year 2010 issues in cryptographic algorithms. In order to successfully complete the transition by 2010, the deadline set by NIST, it is necessary for the financial institutions to begin discussing the issues at the earliest possible date.

This paper summarizes security evaluation results of the current algorithms, and describes Year 2010 issues, their impact to the financial industry, and the transition plan announced by NIST. This paper also shows several points to be discussed when dealing with Year 2010 issues.

Keywords: cryptographic algorithm, symmetric cipher, asymmetric cipher, security, Year 2010 issues, hash function

JEL classification: L86, L96, Z00

† Institute for Monetary and Economic Studies, Bank of Japan (currently Research Center for Information Security, National Institute of Advanced Industrial Science and Technology) (E-mail: masashi-une@aist.go.jp)

* Senior Research Engineer, NTT Information Sharing Platform Laboratories (E-mail: kanda.masayuki@lab.ntt.co.jp)

The authors thank Professor Hideki Imai of Chuo University (Director of Research Center for Information Security, National Institute of Advanced Industrial Science and Technology), Professor Tsutomu Matsumoto of Yokohama National University and Dr. Yiqun Lisa Yin for their valuable comments. Views expressed in this paper are those of the authors and do not necessarily reflect the official views of the Bank of Japan, National Institute of Advanced Industrial Science and Technology, or NTT Information Sharing Platform Laboratories.

Table of Contents

1. Introduction (executive summary)

Cryptographic algorithms are widely used in the financial sector to ensure security in various financial transactions. For instance, the algorithms are used to ensure confidentiality and integrity of data such as a PIN (personal identification number) and an account number in ATM transaction. The algorithms are also used to authenticate counterparties of the transactions in Internet banking services. In general, symmetric ciphers are adopted to ensure the confidentiality. To ensure the integrity and the authenticity, a MAC (message authentication code) based on symmetric ciphers or a digital signature based on asymmetric ciphers is adopted.

To determine which algorithms are currently mainstream, we can refer to international standards and guidelines regarding security techniques in the financial sector. The representative standards and guidelines are developed and maintained by ISO (International Organization for Standardization). It is likely that many financial institutions take security measures in accordance with the ISO standards. For instance, ISO 9564-2 [21], the international standard for PIN encipherment for financial transactions, specifies triple DES as an encryption algorithm for the PIN and also specifies RSA as a key transport scheme for a triple DES session key. ISO 11568-2 [22], the international standard for key management for asymmetric ciphers, specifies several algorithms using SHA-1 as a hash function. As for triple DES, two-key triple DES, which uses two different keys, appears to be the most widely used. RSA with a 1024-bit key (referred to as 1024-bit RSA hereafter) appears to be the most widely used among asymmetric ciphers.

In order to ensure the security of financial services provided through the Internet, the financial institutions use SSL (Secure Sockets Layer)/TLS (Transport Layer Security) specified by IETF (International Engineering Task Force). IETF develops and maintains international standards regarding information techniques used in the Internet. RC4 specified in SSL version 3.0/TLS version 1.0 also appears to be widely used as a symmetric cipher to assure the confidentiality of the transaction data in Internet banking systems.

NIST (National Institute for Standards and Technology) of the United States has approved 2-key triple DES, 1024-bit RSA and SHA-1 as cryptographic algorithms suitable for IT systems of the Federal Government. However, NIST indicated that it will no longer approve these algorithms after 2010. NIST is granted the authority for information security measures for the US Federal Government through various acts and orders. In May 2004, NIST indicated in SP 800-67 (Recommendation for the Triple

Data Encryption Algorithm Block Cipher [52]) that it will no longer approve two-key triple DES. In August 2005, NIST indicated in SP 800-57 (Recommendation on Key Management [46]) that it will no longer approve 1024-bit RSA. As for SHA-1, in August 2004, NIST also announced to phase out SHA-1 by the end of 2010 [51]. As for RC4, NIST has never approved or recommended it.

We can consider the following two reasons for the NIST's decision. The first is the decline of security levels of the conventional algorithms developed in the mid-1990s or before due to recent advances such as progress of cryptanalysis and distributed computation techniques, improvement of computation power and so on. The results of academic security evaluations of the algorithms indicate that they will not be able to provide adequate security within 10 to 15 years. For instance, it was demonstrated that a 663-bit composite number could be factorized practically as of 2005 and 911-bit special composite number was also factorized in 2006. With regard to the hash function, Chinese cryptographers proposed a new and strong method which could find a SHA-1 collision (a pair of different input values that produce the same hash value) more efficiently than the birthday attack[1].

The second is that there exist new algorithms and enhancement to conventional algorithms that provide higher security (and processing speed) than the conventional ones. After NIST's solicitation and selection of AES (the Advanced Encryption Standard), similar activities were taken in Europe and in Japan. In Europe, the NESSIE (the New European Schemes for Signatures, Integrity, and Encryption) project has undertaken a solicitation and selection of recommended cryptographic algorithms (hereafter referred to as NESSIE-recommended cryptographic algorithms [54]). In Japan, CRYPTREC (the Cryptography Research and Evaluation Committees) evaluated cryptographic algorithms and compiled the e-Government recommended ciphers list [45]. Referring to these evaluation results, ISO developed ISO/IEC 18033, an international standard for encryption algorithms [32, 33, 34], for the first time. These evaluations and the international standard help the financial institutions select next-generation algorithms that offer reliable security.

The NIST approval of cryptographic algorithms has served in many fields, including the financial sector, as a guarantee for the security of the corresponding algorithms. For instance, DES became the *de facto* and *de jure* international standard for a symmetric cipher after NIST approved DES as FIPS 46 (Data Encryption Algorithm) in 1977. The NIST guarantee for the current mainstream algorithms will cease to apply after 2010 according to information disclosed by NIST. Therefore, it is required to transit to the

[1] The birthday attack is a basic method of finding a collision on the basis of the birthday paradox. The birthday paradox refers to the fact that probability of at least two people having the same birthday in a group consisting of 23 people exceeds 0.5.

new algorithms which have been evaluated to be sufficiently secure for a long-term period in order to maintain the security and reliability of IT systems in the finance sector. Since such a transition will make each financial institution update its IT systems, it will be also necessary to consider how to ensure compatibility among IT systems at different financial institutions. We can imagine that such consideration necessitates consensus-building in the financial industry, as well as an adequate preparatory period. In addition, it will also take certain costs to update the systems.

It is therefore a critical issue to study the optimal approach to completing a smooth, rapid transition of cryptographic algorithms without damaging security and reliability of IT systems. This paper collectively refers to problems associated with the transition of cryptographic algorithms toward 2010 as Year 2010 issues in cryptographic algorithms (hereafter simply “Year 2010 issues”).

When financial institutions address Year 2010 issues, the first problem is selecting appropriate algorithms. This selection differs from past transitions from DES to triple DES. More secure algorithms have already been reccommended by public organizations in addition to NIST. For instance, CRYPTREC has published the e-Government recommended ciphers list, and NESSIE has published NESSIE-recommended cryptographic algorithms. (See Table 5(4) of the main text.) ISO/IEC JTC1/SC27 has standardized ISO/IEC 18033 on the basis of these recommendations. It is desirable for the financial institutions to select new algorithms by referring to these evaluation results and the international standard.

There are also medium- to long-term issues to be discussed. In order to immediately and appropriately deal with the problems regarding compromise of cryptographic algorithms such as the Year 2010 issues, it is necessary to follow the latest security evaluations of the algorithms and to establish systems to monitor the actions of NIST and other organizations. Financial institutions should also discuss how to design and implement IT systems that can smoothly accommodate changes in cryptographic algorithms and in key lengths.

This paper provides information that will aid financial institutions in examining Year 2010 issues. Section 2 summarizes the cryptographic algorithms currently used in financial institutions, focusing on the international standards and guidelines. Section 3 summarizes current security evaluation results for the cryptographic algorithms described in Section 2. Section 4 discusses NIST policies concerning the transition of the cryptographic algorithms and potential effects of Year 2010 issues. Section 5 introduces cryptographic algorithms such as those specified in ISO/IEC 18033, those approved or recommended by NIST, and those recommended by CRYPTREC or NESSIE, with the goal of determining which cryptographic algorithms should be selected, and discusses various issues to be considered when selecting cryptographic

algorithms and when moving to the selected new cryptographic algorithms, and the problems that financial institutions will need to address with respect to compromise of cryptographic algorithms. Section 6 reviews importance of appropriate responses for the Year 2010 issues and summarizes this paper.

2. Current mainstream cryptographic algorithms

In this section, we will summarize cryptographic algorithms used in the financial sector. We focus on the algorithms specified in international standards and guidelines related to the financial transactions, as well as on those specified in the IETF Internet standards.

(1) Cryptographic algorithms specified or described in international standards and guidelines

In most cases, for security reasons, financial institutions do not disclose the cryptographic algorithms used in their networks and IT systems. For this reason, we refer to the international standards and guidelines maintained by ISO/TC68, which is responsible for the development and management of international standards for information security techniques in the financial sector. Table 2 (1) summarizes the algorithms which are specified or described in these standards and guidelines.

Table 2 (1) clearly shows that triple DES is the most widely used symmetric cipher. ISO 9564-1 [20] specifies that the key length should be at least 112 bits. No other standards or guidelines explicitly restrict descriptions to two-key triple DES, but they also include three-key triple DES. AES is described as a recommended algorithm only in ISO/TR 17944 (financial system security framework [26]).

All ISO/TC68-related international standards for asymmetric ciphers specify RSA. The only description for the recommended key length of RSA occurs in ISO/TR 13569, which recommends 1024 bits or longer [25]. Other than RSA, ISO/TR 17944 describes DSA and ECDSA as recommended algorithms. According to ISO/TR 13569, DSA and ECDSA key lengths should be 1024 bits or longer and 160 bits or longer, respectively.

All ISO/TC68-related international standards that address hash functions specify SHA-1. ISO/DIS 11568-4 [23], a draft for the international standard for key management using asymmetric ciphers, specifies hash functions specified in the ISO/IEC 10118 (the international standards of hash functions for general commerce) [28] as the approved hash functions in the normative annex. ISO/IEC 10118 also specifies SHA-1.

Several technical specifications are also widely referenced. These include the Japanese Bankers Association IC cash card specifications [37], EMV smart card specifications (version 4.1 [11]), and FINREAD technical specifications for smart card readers [13]. These specifications also exhibit the same tendencies as the international standards. It is especially notable that EMV specifies two-key triple DES for a symmetric cipher.

Table 2 (1) Cryptographic algorithms specified in ISO/TC68 international standards

International standards and guidelines	Symmetric ciphers	Asymmetric ciphers	Hash functions
ISO 10126-2 (New Proposal) - Message encryption	DES	(No description)	(No description)
ISO 16609 (MAC requirements)	• DES • TDES(*1)	(No description)	(No description)
ISO 11568-2 - Key management (Symmetric ciphers)	TDES	(No description)	(No description)
ISO/DIS 11568-4 — Key management (Asymmetric ciphers)	(No description)	• RSA (ISO/IEC 9796) • DSA	ISO/IEC 10118 • 10118-2: Specifies 4 methods based on symmetric ciphers • 10118-3: RIPEMD-(128, 160), SHA-(1, 224, 256, 384, 512), Whirlpool • 10118-4: MASH-1,2
ISO TR 17944 — Security management framework	• TDES (ANS X9.52) • AES (FIPS 197)	• RSA (ANS X9.31) • DSA (ANS X9.30-1) • ECDSA (ANS X9.62) • ISO/IEC 15946 • ISO/IEC 9796 • ISO/IEC 14888	
ISO 9564-1, 2 — PIN encryption	TDES (*2)	RSA（EMV）	(No description)
ISO TR 19038 — TDES modes of operation	TDES	(No description)	(No description)
[Reference] EMV version 4.1	2-key TDES	RSA (ISO/IEC 9796-2)(*3)	SHA-1
[Reference] Japanese Bankers Association (JBA) IC Cash Card Standard Specifications (2006, recommended)	• DES • TDES (In compliance with EMV, although not explicitly stated as two-key)	RSA (*4)	• SHA-1
[Reference] FINREAD	• DES • TDES	RSA (*5)	• SHA-1 • MD5 • RIPEMD-160

Notes:
(*1) “TDES” stands for “triple DES.”
(*2) ISO 9564-1 specifies that PIN encipherment keys shall be at least 112 bits in length [20].
(*3) EMV specifies 1984 bits as the maximum key length for RSA [11].
(*4) JBA specifies 1984 bits as the maximum key length for RSA [37].
(*5) FINREAD specifies 1024 bits or longer as the length for RSA [13].

Since it is likely that financial institutions actually adopt cryptographic algorithms in accordance with these international standards and guidelines, it is reasonable to assume that triple DES is widely used as a symmetric cipher in the financial sector.

Published documents do not clearly indicate which of two-key or three-key triple DES are adopted. However, since ISO 9564-1 specifies the key length used to encrypt PINs as at least 112 bits, and as many applications adopt EMV specifying use of two-key triple DES, as with the IC cash card standard specifications of JBA, one naturally would

conclude that two-key triple DES is used quite widely.

Descriptions of RSA key length are also scarce, with only ISO/TR 13569 specifying a key length of 1024 bits or longer. In order to implement RSA efficiently, it is reasonable to select the shortest key length, 1024 bits. The SWIFT BKE (bilateral key exchange) is a well-known example of 1024-bit RSA [60]. BKE is a system for distributing keys for generating a MAC in SWIFTNet, which is the network system for transmitting transaction information between financial institutions. SWIFT has openly indicated that it uses 1024-bit RSA as the key transport algorithm. EMV security guidelines also give the recommended RSA key length implemented in smart cards as 1024 bits, up to the end of 2009 [12]. Here, these guidelines also state the recommended key lengths for use up to the end of 2012, 2014, and 2016 as 1152 bits, 1408 bits, and 1984 bits, respectively.

(2) Cryptographic algorithms specified in the IETF standards

When providing Internet banking services, financial institutions use cryptographic algorithms to authenticate their customers (client authentication) and to encrypt transmitted data. These encryption and authentication functions are generally implemented by the SSL (Secure Sockets Layer), a *de facto* standard in web browsers such as Internet Explorer. SSL is specified in the RFC of IETF (Internet Engineering Task Force) as the TLS (Transport Layer Security) version 1.0 [3, 8]. An Internet draft of TLS version 1.1 has also been proposed [4, 9].

The cryptographic algorithms used in SSL version 3.0/TLS version 1.0 are as follows: The symmetric ciphers are triple DES, DES, RC2, RC4, IDEA, AES, Camellia, and SEED; asymmetric ciphers are RSA, DSA, and the Diffie-Hellman key agreement scheme (referred to as DH hereafter); and hash functions are SHA-1 and MD5. Among these, triple DES and RC4 are considered mainly used for symmetric ciphers. The key length for an asymmetric cipher is generally assumed to be 1024 bits in most cases (e.g., [56]).

Other cryptographic algorithms specified in the RFC of IETF include MISTY1 for symmetric ciphers and ECDH and ECDSA for asymmetric ciphers.

3. Security evaluation results for major cryptographic algorithms

Section 3 summarizes security evaluation results for the cryptographic algorithms which were described in Section 2.

(1) Symmetric ciphers

Symmetric ciphers can be classified into block ciphers and stream ciphers. Block ciphers divide a plaintext to be encrypted into portions of a fixed size (The portions are referred to as “blocks.”) and encrypt each block in a time. Stream ciphers generate pseudo-random numbers of the same size as the plaintext and generate the ciphertext by calculating the exclusive OR serially bit by bit. The security evaluation results for the cryptographic algorithms introduced in Section 2 are described below: namely, triple DES, DES, RC2, IDEA, MISTY1, AES, Camellia, SEED (these are block ciphers), and RC4 (a stream cipher), according to these categories.

A. Block ciphers

Attacks against the block ciphers can be divided into shortcut attacks and brute force attacks. The shortcut attacks try to minimize the computational complexity required to find the correct key by exploiting the analytical and statistical characteristics of the algorithms. The brute force attacks try one possible encryption key after another to obtain information on the correct key and/or the plaintext. We define that a block cipher is “academically broken” by a certin attack if the computational complexity required for the attack is less than that for an exhaustive search[2]. We call a block cipher “secure” if such an attack has not been found so far. The situation of “academically broken” does not always mean that the algorithm cannot be used for practical operations immediately. However, the fact of being academically broken will reduce confidence in the performance of a cryptographer who designed the corresponding algorithm and raise possibility of the other yet-to-be-discovered fatal defects. Consequently, its use will tend to decline.

(A) Shortcut attacks

[2] A type of brute force attack, the exhaustive key search tries to find the correct key by trying one candidate key after another. For a cipher with key length of n bits, the exhaustive key search needs to perform 2^n encryption operations in order to find the key with a probability of 1.

Table 3 (1): Security of block ciphers against shortcut attacks

Cryptographic algorithms	Key length	Block length	Results of shortcut attacks
Triple DES (two-key/three-key)	112 or 168	64	[Secure] No reports exist of shortcut attacks capable of finding the correct key with computational complexity less than that required for the exhaustive key search.
RC2	Variable (40 in SSL v.3.0/TLS v.1.0)	64	[Academically broken] There exists the differential cryptanalysis capable of finding the correct key with computational complexity less than that required for the exhaustive key search.
IDEA	128	64	[Secure] No reports exist of shortcut attacks capable of finding the correct key with computational complexity less than that required for the exhaustive key search.
MISTY1	128	64	
AES	128, 192, 256	128	
Camellia	128, 192, 256	128	
SEED	128	128	

Note: This table is based on information available as of February 20, 2006.

The applicability of the shortcut attacks depends on the structure of each algorithm. Table 3 (1) summarizes information on the shortcut attacks proposed to date for each algorithm. The table shows that RC2 has been academically broken with the differential cryptanalysis[3] [36]. No strong shortcut attacks are reported for the other algorithms.

(B) Brute force attacks

A cryptographic algorithm is also evaluated as to whether it is academically broken by brute force attacks by comparing the computational complexity required to perform a specific attack in question with that required for the exhaustive key search. Table 3 (2) summarizes the results of applying brute force attacks.

For triple DES, both two-key and three-key triple DES has already been academically broken with lower computational complexity (2^{57} and 2^{112}, respectively) than that required for the exhaustive key search [44]. Since this attack requires obtaining 2^{56} pairs of plaintext and ciphertext, such an attack is not expected to pose problems in practice. As the results, however, two-key triple DES is now evaluated in respect of computational complexity as “it is permissibly said that 2-key Triple DES can be practically broken because the number of calculation complexity is two times that of the exhaustive key search” according to CRYPTREC Report 2002 (p. 172 in [36]). On the other hand, since the block size (64 bits) of triple DES

[3] The differential cryptanalysis is a class of attacks that can efficiently find the correct key by using algorithmic characteristics that a certain difference of ciphertext pairs occurs under a certain difference of plaintext pairs with probability higher than 0.5.

Table 3 (2): Security of block ciphers against brute force attacks

Cryptographic algorithms	Key length	Block length	Results of brute force attacks
Two-key triple DES	112	64	[Academically broken] An attack capable of finding the correct key with computational complexity (2^{57}), which is less than that required for the exhaustive key search, has been proposed. Although it requires 2^{56} pairs of plaintexts and ciphertexts, **the computational complexity is reaching a feasible level.** [Note] A ciphertext matching attack is possible with approximately 2^{32} ciphertexts.
Three-key triple DES	168	64	[Academically broken] An attack capable of finding the correct key with computational complexity (2^{112}) less than that required for the exhaustive key search has been proposed. It requires 2^{56} pairs of plaintexts and ciphertexts. [Note] A ciphertext matching attack is possible with approximately 2^{32} ciphertexts.
RC2	Variable (40 in SSL v.3.0/TLS v.1.0)	64	[Note] No reports have been published on attacks capable of finding the correct key with computational complexity less than that required for the exhaustive key search. However, a ciphertext matching attack is possible with approximately 2^{32} ciphertexts.
IDEA	128	64	
MISTY1	128	64	
AES	128, 192, 256	128	[Secure] No reports exist of attacks capable of finding the correct key with computational complexity less than that required for the exhaustive key search.
Camellia	128, 192, 256	128	
SEED	128	128	

Note: This table is based on information available as of November 7, 2005.

is relatively short, its potential susceptibility to a ciphertext matching attack has been pointed out.[4] The memory required to apply such an attack to triple DES is 32 gigabytes, which is increasingly becoming feasible.

For RC2, IDEA, and MISTY1, no brute force attacks enabled with lower computational complexity than the exhaustive key search have been reported. However, as with triple DES, these ciphers require consideration with respect to risks of the ciphertext matching attack.

No such concerns currently apply for 128-bit block ciphers, AES, Camellia, and SEED. The ciphertext matching attack for a 128-bit block cipher requires at least 2^{64} ciphertexts encrypted with the same key and 2^{28} terabytes of memory. Given the difficulty of preparing this enormous amount of memory, these ciphers are

[4] The ciphertext matching attack collects a large number of ciphertexts encrypted with the same key, searches for the same ciphertexts, and uses the result to find the corresponding plaintexts and correct key. Security against this attack can be evaluated based on block size. For example, for a *n*-bit block cipher, if $2^{n/2}$ ciphertexts (approximately $n*2^{(n/2-33)}$ gigabytes) encrypted by the same key are randomly collected, it is known that at least a pair of ciphertexts can be found with probability higher than 0.5.

considered adequately secure against the ciphertext matching attack.

B. Stream ciphers

Stream ciphers basically generate a pseudo-random number whose size is the same as that of a plaintext and calculate exclusive-or between the pseudo-random number and the plaintext bit by bit to generate a ciphertext. Its security depends largely on the pseudo-random number generator. If the generator has a defect whereby the encryption key and the pseudo-random numbers are strongly correlated or future pseudo-random numbers can be efficiently predicted from the past numbers, an attacker can easily obtain the pseudo-random numbers or the correct key and, as the results, decrypt the ciphertext.

RC4 is considered to be the most widely used among stream ciphers proposed to date. SSL version 3.0/TLS version 1.0 also adopts RC4 as one of its symmetric ciphers, and many financial institutions appear to select RC4 as the symmetric cipher for data encryption in Internet banking systems.

According to the security evaluation results for RC4, no attacks that academically break RC4 have been reported as long as the key length is 128 bits and the parameter[5] is set as being specified in the standard specifications in SSL version 3.0/TLS version 1.0. However, for the RC4 implementation incorporated into WEP (Wired Equivalent Privacy) which is the communication protocol used in wireless LANs, strong correlation may arise between the pseudo-random numbers and the encryption key, due to insufficient permutation of initial states. An example is known of successful identification of the encryption key based on this weakness [15][6]. Thus, CRYPTREC does not recommend RC4 with parameters other than those specified in the SSL version 3.0/TLS version 1.0 standard [45].

(2) Asymmetric ciphers

This paper selects RSA, DSA, DH, and ECDSA as examples of asymmetric ciphers and summarizes security evaluation results for these algorithms[7]. We define “probable

---

[5] This parameter determines internal states. For example, when the value of the parameter is n, the number of internal states is 2^n. The standard parameter setting is $n = 8$.

[6] Researchers have devised ways to improve the security of WEP against such attacks by modifying part of WEP, given WEP's wide deployment. However, some researchers believe even these measures are inadequate [66].

[7] It is known that factoring could be readily solved if quantum computers would be developed. However, some researchers argue that factoring several-thousand-bit public keys would require a quantum computer with several tens of thousand q-bits. The development of such a machine within 20 to 30 years is

security" as a property that security of a cryptographic algorithm can be proved to be equivalent[8] to the difficulty of a mathematical problem such as factoring under certain assumptions.

A. RSA (Evaluation of difficulty in factoring)

The security of RSA depends on the difficulty of factoring of large composite numbers. The RSA primitive algorithm permits numerous variations in encryption and digital signature schemes. For example, the encryption schemes include PKCS#1 version 1.5 (RSAES-PKCS1-v1_5), RSA-OAEP and RSA-KEM, while the signature schemes include RSA-PSS and the ISO/IEC 9796. These schemes have been proposed in order to provide the RSA primitive algorithm with specific security features and/or to improve the performance. For example, some among them, such as RSA-OAEP, RSA-KEM and RSA-PSS, provide provable security - i.e., the security of the scheme can be proved to be equivalent to the difficulty of factoring under certain assumptions. However, none of these methods could prevent cyptanalysis and signature forgery either if an efficient algorithm for factoring were proposed or if a powerful hardware for factoring were developed.

Many researchers have performed computer experiments to determine the size of the composite number (corresponding to RSA key length) that can be practically factored. In May 2005, with the general number field sieve, which is currently considered the fastest, a 663-bit composite number was successfully factored[9].

Several papers have also been published on how to build dedicated factoring hardware. For instance, [16] proposed SHARK which was the dedicated hardware for the sieving process (also referred to as collection of relations). The sieving process accounts for the largest part of the general number field sieve in terms of the computational complexity. Their results indicate that the sieving process for the 1024-bit factoring can be performed at a cost of approximately 200 million dollars over a period of one year. On the other hand, [18] discusses feasibility of dedicated hardware that processes matrix calculations with regard to factoring of a 1024-bit composite number. They claimed that the matrix calculations could be done at a cost of approximately 2 million dollars over a period of approximately 2.4 months.

Discussed next are the results of studies on the feasibility of factoring of a 1024-bit

---

considered highly unlikely. Thus, we omit considering the implications of the quantum computers for security in the ensuing discussion.

[8] This means that the algorithm can be broken if and only if the mathematical problem can be solved.

[9] In January 2006, a 911-bit special composite number was also successfully factored.

composite number with respect to the computational complexity and cost. Brent examined the possibility of future factoring from the past achievements based on Moore's Law [5][10]. He estimated that a 1024-bit composite number may be able to be practically factored somewhere around the year 2018 with the general number field sieve. Lenstra and Verheul examined the key length required to achieve strength equal to as DES in 1982, at which DES was regarded as being adequately secure [40]. They concluded that 1024-bit RSA in 2002 approximately offered the same security level as DES in 1982. It recommends a RSA key length of 2048 bits to provide adequate security for 20 years beginning 2001. Kaliski recommended 1024-bit RSA as an asymmetric cipher providing adequate security until 2010, after which he recommended moving to 2048-bit RSA or higher for use up to 2030 [38].

The NESSIE report on the cryptographic algorithm evaluation [56] showed results of studies on the computational complexity for ensuring medium-term security. It is assumed that the computational complexity required for 512-bit factoring is equivalent to that required for the exhaustive key search for a 56-bit key, i.e., 2^{56} of computational complexity. Based on this assumption, the report concluded that the computational complexity required for factoring of a 1536-bit composite number is approximately equivalent to that required for the exhaustive key search for a 80-bit key.

According to these results, we can say that a 1024-bit RSA will not provide adequate long-term security in 2010.

B. DSA and DH (Evaluation of difficulty in discrete logarithm problem)

DSA does not provide the provable security, but no fatal defects have been published to date with respect to the algorithm itself, to the best of the authors' knowledge, except for the weakness in the pseudo-random number generator. As for DH, no efficient attack against the algorithm itself has been proposed, again, to the best of the authors' knowledge, except to solve the discrete logarithm problem.

The security of DSA and DH depends on the difficulty of solving the discrete logarithm problem in the multiplicative group of a finite field (hereafter referred to as DLP). Currently, the fastest algorithm for solving DLP is the index calculus, and various methods have been proposed as its variations (e.g., [7, 10, 19, 55, 58, 59]). In 2002, it was reported that DLP in the multiplicative group with 607-bit modulus (the

[10] Moore's law, proposed by Gordon Moore, states that integration density of a semiconductor device doubles every 18 to 24 months. This law is often used to predict the future performance of semiconductor devices and the development of associated information technologies. The idea often arises in the context of information security.

modulus corresponds to the key length) could be successfully solved [62].

The algorithm for the index calculus is known to be closely related to the general number field sieve for factoring. The order of the computational complexity required to solve DLP with the index calculus is regarded as being equivalent to that required for the general number field sieve when key length is the same (e.g., [56]). Thus, the key lengh of the algorithms based on DLP is generally set as long as that of the algorithms based on the factoring problem. In fact, as with RSA, 1024-bit key length is generally adopted in DSA and DH. As described in subsection A of this section, it is likely that factoring of a 1024-bit composite number will no longer be infeasible for a long time period as of 2010. Thus, the key length of DSA and DH should be updated as 2010 approaches.

C. ECDSA (Evaluation of difficulty in elliptic curve discrete logarithm problem)

ECDSA is based on the difficulty of solving the discrete logarithm problem for the group of rational points on an elliptic curve over some finite field (hereafter referred to as ECDLP, the elliptic curve discrete logarithm problem). As with DSA, ECDSA lacks the provable security, but the authors know of no fatal attack capable of breaching the security. For this reason, in current evaluations of ECDSA security, we focus on the relationship between the difficulty of ECDLP and the key length.

It is known that the index calculus cannot be easily applied to solve ECDLP for ECDSA if the elliptic curve is appropriately selected. The most efficient algorithm for solving ECDSA differs with elliptic curves used. This implies that it is necessary to avoid selecting specific elliptic curves under which ECDLP can be efficiently solved. Such curves can be found in [17, 43, 57]. ANS X9.62 (which ISO/TR 17944 also refers), which specifies ECDSA for financial purposes, does not recommend any specific elliptic curves.

Given these concerns, the question is to determine an appropriate key length. Since ISO/TR 13569 specifies that the key length (corresponding to the size of the order of the finite field) should be set as 160 bits or more, 160 bits appears to be selected in the financial sector.

The computational resources and time required to solve ECDLP have been demonstrated in a contest held by Certicom Inc. ECDLP over the finite field with the order of 109 bits has reportedly been solved.

For ECDLP over the finite field with the order of 160 bits, Lenstra and Verheul estimated that, in 2010, elliptic curve cryptosystems with a 160-bit key would ensure security equivalent to DES in 1982 – i.e., they would provide adequate long-term security – under an assumption that cryptanalytic progress halves the computational

complexity every 18 months [40]. When the assumption does not hold, they see the level of security provided in 2020 as equivalent to DES in 1982. According to the NESSIE report, the key length required to ensure medium-term security was estimated as 160 bits as of 2003. This conclusion is consistent with the key length currently adopted. On the other hand, Certicom Research estimated that elliptic curve cryptosystems with a 160-bit key provided security equivalent to 1024-bit RSA and DSA or symmetric ciphers with a 80-bit key [6].

According to Lenstra and Verheul, it is likely that ECDSA with a 160-bit key does not present particular problems after 2010 [40]. According to Certicom Research, the key length should be lengthened to 224 bits or more [6]. As such, there exists a range of the minimum recommended key length between 160 and 224 bits. If one would like to pay much attention to the security, it is desirable to select the key length of 224 bits or more as reccommeded in Certicom Research.

(3) Hash function

In this subsection, we will focus on SHA-1 because it is the most widely used hash function. In February 2005, Wang, Yin and Yu published research results that SHA-1 collisions[11]can be found with the computational complexity equivalent to 2^{69} hash function operations [65]. In addition, Wang, Yao and Yao also claimed that SHA-1 collisions can be found with the computational complexity equivalent to 2^{63} hash function operations [64][12]. However, no examples of message pairs which cause collisions have been published.

To find collisions with relatively high probability for a secure hash function, one must collect $2^{(n/2)}$ hash values when the size of the hash values is n bits because of the birthday attack. If it becomes clear that collisions can be found with the computational complexity less than $2^{(n/2)}$ hash function operations, the hash function is regarded as being academically insecure. For SHA-1, the size of the hash values is 160 bits, which makes $2^{(n/2)} = 2^{80}$. This amount is larger than 2^{69} which is the computational complexity required for Wang's attack. This means that SHA-1 is academically insecure if Wang's claim is correct.

Even though collisions may have been found, they would rarely occur in meaningful messages. In addition, if a message would be altered using the collisions, it would be likely that the alteration could be acturally detected by checking the meaning of the

---

[11] A collision in a hash function refers to a pair of different input values that produce the same output. Hash functions are designed so that it is difficult to find such collisions.

[12] In RSA Conference 2006, Wang showed a new estimation that SHA-1 collisions can be found with the computational complexity equivalent to about 2^{61} hash function operations [63].

message. In other words, the hash function has not been rendered instantly insecure for real-world use, and practical attacks in which the alteration cannot be detected even when considering the meaning of the sentences still present formidable difficulties.

Nevertheless, one cannot always conclude that “academically insecure” hash functions are free from serious security breaches. If data added to messages such as control codes, padding data, and random number sequences can be altered, collisions may arise between two different messages unrelated to the sentences themselves or whose validity cannot be determined from the meaning of the sentences [42]. In these cases, it is difficult to detect alterations even when considering the meaning of the sentences.

Lenstra, Wang and de Weger proposed a method for the construction of pairs of valid X.509 certificates in which the “to be signed” parts form a collision for the MD5 hash function [41]. As a result, the issuer signatures in the certificates will be the same when the issuer uses MD5 as its hash function. Although the attack is not practical, their paper shows potential applications of collisions of hash functions.

The potential for a serious breach resulting from collisions depends on how the hash function is used. To ensure the utmost security, it is generally considered to be appropriate to switch to a new hash function which is evaluated to be more secure if the current hash function is linked even once to the possibility of a collision.

4. NIST policies and impacts on Year 2010 issues

This section addresses current NIST policies regarding a transition of the cryptographic algorithms discussed in the previous section, as well as Year 2010 issues.

(1) NIST policies in selecting cryptographic algorithms

A. FIPS and SP

NIST is granted the authority for information security measures for the US Federal Government through various acts and orders, including the Computer Security Act of 1987, the Information Technology Management Reform Act of 1996, the Federal Information Security Management Act of 2002, and Executive Order #13011. Based on this authority, NIST develops FIPSs[13] and SPs[14], which involve cryptographic techniques, security products, their evaluation methods, and security management. (See Table 4 (1).)

For cryptographic algorithms, NIST has developed FIPSs and SPs that specify block ciphers, digital signature schemes, and hash functions. For block ciphers, NIST has approved AES (FIPS 197), two-key/three-key triple DES (SP 800-67), and Skipjack (FIPS 185). For digital signature schemes, NIST has approved RSA, DSA, and ECDSA (FIPS 186-2)[15]. For hash functions, NIST has approved SHA series (SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512) (FIPS180-2). For key agreement schemes, NIST has approved DH and the Menezes-Qu-Vanstone method (referred to as MQV hereafter) (SP 800-56).

With regard to triple DES, FIPS 46-3 which specified triple DES has already been withdrawn in May 2005, and triple DES is thereafter specified in SP 800-67 only.

---

[13] A FIPS (Federal Information Processing Standard) specifies information techniques to be adopted in IT systems used by the US Federal Government (except national security systems) for treating "unclassified but sensitive information" (such as privacy-related information). Security products not conforming to FIPS fail to meet the specification requirements for the systems, and it is virtually impossible to procure such products. For this reason, the cryptographic techniques specified in FIPS are referred to as enforceable US Government standard.

[14] A SP (Special Publication) is published as general technical information for recommended techniques or as accompanying information for a FIPS on an as-needed basis. While SPs are not enforced and adoption is voluntary, based on circumstances. Nevertheless, when they provide information supplemental to a FIPS, an SP tends to describe additional information for specifications and guidelines not specified in the corresponding FIPS. In such cases, the SP may become part of a *de facto* enforceable specification.

[15] In March 2006, NIST published a draft of FIPS 186-3 [48] as a revised version of FIPS 186-2.

Table 4 (1) Major FIPSs and SPs concerning cryptographic techniques

Type	Specification number	Title	Date of issue
Implementation policies	SP 800-21	Guideline for Implementing Cryptography in the Federal Government	November 1999
Key management	SP 800-57	Recommendation on Key Management	August 2005
Cryptographic algorithms for personal identity verification and corresponding key length	SP 800-78	Cryptographic Algorithms and Key Sizes for Personal Identity Verification	April 2005
Block ciphers and modes of operation	FIPS 197	Advanced Encryption Standard (AES)	November 2001
	SP 800-67	Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher	May 2004
	FIPS 185	Escrowed Encryption Standard (EES)	February 1994
	SP 800-38A	Recommendation for Block Cipher Modes of Operation – Methods and Techniques	December 2001
	SP 800-38B	Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication	May 2005
	SP 800-38C	Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality	May 2004
	SP 800-38D	Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) for Confidentiality and Authentication (Draft)	April 2006
Digital signature	FIPS 186-2*	Digital Signature Standard (DSS)	January 2000
Hash function	FIPS 180-2	Secure Hash Standard (SHS)	August 2002
Key agreement scheme	SP 800-56A	Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography	March 2006
Message authentication	FIPS 113	Computer Data Authentication	May 1985
	FIPS 198	The Keyed-Hash Message Authentication Code (HMAC)	March 2002
Entity authentication	FIPS 196	Entity Authentication Using Public Key Cryptography	February 1997
Pseudorandom number generation	SP 800-90	Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Draft)	December 2005
Password use and generation	FIPS 181	Automated Password Generator	October 1993

Note: A draft of FIPS 186-3 that will supersede FIPS 186-2 is available for public comment in March 2006.

The SP states that two-key triple DES may be used up to 2010 but not thereafter.

B. Prospects for transition of cryptographic algorithms and key lengths

In addition to approval of cryptographic algorithms through FIPSs and SPs, NIST also releases SP 800-57 and SP 800-78 guidelines that provide information on the transition prospects for the algorithms and key lengths.

Table 4 (2) Prospects of transitions in cryptographic algorithms described in SP 800-57

Algorithm security lifetime	Symmetric ciphers	Asymmetric ciphers based on factoring	Asymmetric ciphers based on DLP	Asymmetric ciphers based on ECDLP
Through 2010	• 2-key triple DES • 3-key triple DES • AES (with key length of 128, 192, or 256 bits)	Minimum key length of 1024 bits	Minimum key length of 1024 bits (order q of the subgroup over the finite field is 160 bits.)	Minimum key length of 160 bits
Through 2030	• 3-key triple DEA • AES (with key length of 128, 192, or 256 bits)	Minimum key length of 2048 bits	Minimum key length of 2048 bits (order q of the subgroup over the finite field is 224 bits.)	Minimum key length of 224 bits
Beyond 2030	AES (with key length of 128, 192, or 256 bits)	Minimum key length of 3072 bits	Minimum key length of 3072 bits (order q of the subgroup over the finite field is 256 bits.)	Minimum key length of 256 bits

Note: The table is summarized based on Table 4 from [46].

(A) SP 800-57

SP 800-57 is a guideline which addresses key management of the cryptographic algorithms used in IT systems of the Federal Government [46]. This provides detailed information required to implement the cryptographic algorithms, including various types of encryption keys, their use, and required key lengths (See Table 4 (2)). As shown in Table 4 (2), SP 800-57 describes NIST recommendation of cryptographic algorithms and key lengths by separating cases into those for which they may be used up to the end of 2010, up to the end of 2030, and after 2030.

For applications approved to up to the end of 2010, the recommended symmetric ciphers include two-key triple DES. However, two-key triple DES is not included for applications to be used after 2010. For use of asymmetric ciphers based on the factoring problem and DLP through 2030, the recommended key length is 2048 bits. For those based on ECDLP, the recommended key length is 224 bits.

For two-key triple DES, 1024-bit RSA and 1024-bit DSA, which are believed to be in wide use in the financial sector, NIST has indicated that the Federal Government will drop their use from its IT systems after 2010.

SP 800-57 states that it does not recommend SHA-1 as a hash function used for digital signature applications when constructing new IT systems in the future.

Table 4 (3) Prospects of transition of cryptographic algorithms described in SP 800-78

Key type	Time period for use	Recommend cryptographic algorithms and key lengths	
A mandatory PIV authentication key (Asymmetric ciphers)	Through 2010	• RSA (Key length: 1024, 2048, or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)	
	After 2010	• RSA (Key length: 2048 or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)	
An optional card authentication key (Symmetric ciphers or asymmetric ciphers)	Through 2010	Symmetric ciphers	• two-key triple DES • three-key triple DES • AES (Key length: 128, 192, or 256 bits)
		Asymmetric ciphers	• RSA (Key length: 1024, 2048, or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)
	After 2010	Symmetric ciphers	• three-key triple DES • AES（(Key length: 128, 192, or 256 bits)
		Asymmetric ciphers	• RSA (Key length: 2048 or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)
An optional digital signature key	Through 2008	• RSA (Key length: 1024, 2048, or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)	
	After 2008	• RSA (Key length: 2048 or 3072 bits) • ECDSA (Key length: from 224 to 283 bits)	
An optional key management key (Asymmetric ciphers)	Through 2008	• RSA (Key length: 1024, 2048, or 3072 bits) • ECDH or ECC MQV (Key length: from 224 to 283 bits)	
	After 2008	• RSA (Key length: 2048 or 3072 bits) • ECDH or ECC MQV (Key length: from 224 to 283 bits)	

Note: This table is summarized based on Table 3-1 of [47].

(B) SP 800-78

SP 800-78 describes various cryptographic algorithms and key lengths to be adopted in PIV (Personal Identity Verification) systems by the Federal Government [47]. The PIV systems authenticate Federal employees and contractors using official certificates or related information for the purpose of access control to IT systems of the Federal Government. FIPS 201 (Personal Identity Verification for Federal Employees and Contractors) describes configuration of the PIV systems in which smart cards (known as the PIV cards) are distributed to the Federal employees [49]. In FIPS 201, it is illustrated that individuals are authenticated with cryptographic techniques based on the public key infrastructure. SP 800-78 contains information that supplements FIPS 201 in light of cryptographic algorithms and is regarded as a mandatory specification in practice.

SP 800-78 describes the four types of keys stored on the PIV card: (1) a mandatory PIV authentication key, (2) an optional card authentication key, (3) an optional digital signature key, and (4) an optional key management key. The mandatory PIV

authentication key is a secret key for generating a digital signature of the corresponding PIV card holder. The optional card authentication key is a secret key for generating a MAC or a digital signature of the corresponding PIV card. The optional digital signature key is a secret key for generating a digital signature except for the purpose of the card holder authentication.

Table 4 (3) shows cryptographic algorithms and key lengths used for these keys. AES and triple DES are recommended for symmetric ciphers. RSA and ECDSA (or other elliptic curve cryptosystems) are recommended for asymmetric ciphers.

Focusing on the mandatory PIV authentication key in Table 4 (3), we can clearly understand that 1024-bit RSA is recommended for use up to the end of 2010, while 2048-bit RSA is at least recommended after 2010. For symmetric ciphers, two-key triple DES is recommended as the optional card authentication key up to the end of 2010. ECDSA, which is suitable for smart cards, is also included among the digital signature algorithms in addition to RSA. The key length of ECDSA is recommended to be 224 bits or more after 2010.

For hash functions, SP 800-78 recommends SHA-1, SHA-224, and SHA-256 up to the end of 2010, and also recommends SHA-224 and SHA-256 after 2010.

C. Prospects for SHA-1

NIST released a comment concerning the handling of SHA-1 directly after a collision was found for SHA-0, which is SHA-1's predecessor [51]. NIST states as follows: "The results presented so far on SHA-1 do not call its security into question. However, due to advances in technology, NIST plans to phase out of SHA-1 in favor of the larger and stronger hash functions (SHA-224, SHA-256, ASHA-384 and SHA-512) by 2010." And then, in response to Wang's presentation of the attack proposed for SHA-1, NIST released the following comment [53]: "Due to advances in computing power, NIST already planned to phase out SHA-1 in favor of the larger and stronger hash functions (SHA-224, SHA-256, SHA-384 and SHA-512) by 2010. New developments should use the larger and stronger hash functions." In March 2006, NIST also released the following comment at its website[16]: "The SHA-2 family of hash functions (i.e., SHA-224, SHA-256, SHA-384 and SHA-512) may be used by Federal agencies for all applications using secure hash algorithms. Federal agencies should stop using SHA-1 for digital signatures, digital time stamping and other applications that require collision resistance as soon as practical, and must use the SHA-2 family of hash functions for these applications after 2010."

---

16 The URL is http://csrc.nist.gov/CryptoToolkit/tkhash.html.

These comments are consistent with descriptions concerning use of hash functions in SP 800-78 as discussed in subsection B. (B). Thus, it is quite likely that use of SHA-1 will be stopped in the Federal Government by 2010. NIST also held a hash function workshop in October 2005 and plans to hold another one in August 2006 in response to the SHA-1 attack.

(2) Past responses in ISO/TC68: DES and triple DES

As introduced in Section 3 and Section 4 (1), the major evaluation results have indicated that the cryptographic algorithms now widely used in the financial sector are unlikely to maintain adequate security for a long time period. NIST has published the plan for the transition to stronger algorithms in IT systems of the Federal Government by 2010. This indicates the revocation of NIST guarantee for the security of the mainstream algorithms. As the result, the revocation may considerably damage reputation for the security of financial IT systems adopting the mainstream algorithms.

In the financial sector, the international standards were traditionally set based on cryptographic algorithms evaluated by NIST as sufficiently secure. The following two situations similar to Year 2010 issues have arisen before.

The first case involves the approval of DES as FIPS 46 in 1977. After FIPS 46 was released, ISO/TC68 began developing international standards for information security techniques based on DES. It is believed that ISO/TC68 decided to adopt DES in the international standards because NIST had approved DES in FIPS 46 [61].

The second case involves the withdrawal of DES in 1999 and subsequent NIST approval of triple DES as FIPS 46-3. It was considered that Triple DES was approved as a temporal successor of DES until AES was practically available. After the approval of DES as FIPS 46, ISO/TC68 accordingly followed the research results on the security of DES. In 1997, a survey paper regarding the security evaluation of DES [39] was submitted to ISO/TC68 from Japan as a technical contribution. When NIST released FIPS 46-3 in 1999 and determined the transition from DES to triple DES, ISO/TC68 immediately launched discussion on adopting triple DES to its international standards. At the time, ANS X 9.52, which was the US domestic standard developed in 1998, had already been published as the standard for triple DES in the financial sector. NIST released FIPS 46-3 to specify triple DES, referring to ANS X 9.52. Subsequently, ISO/TC68 began discussing possibility of making ANS X 9.52 an international standard. As the results, ISO/TC68/SC2/WG11 developed ISO/TR 19038 which is a technical report regarding how to implement triple DES in 2004. In addition, ISO/TC68 has updated various standards in such a way to transit from DES to triple DES.

Thus, the international standards regarding DES and triple DES in ISO/TC68 have been

developed in accordance with NIST approval of cryptographic algorithms.

(3) Responses to Year 2010 issues and their impact to the financial sector

Year 2010 issues are raised by the fact that the guarantee for the security of cryptographic algorithms will be lifted by NIST around 2010. According to the history of the international standards in the financial industry, financial institutions have placed their trust in the security evaluation results for cryptographic algorithms performed by NIST, and have moved to new cryptographic algorithms in accordance with NIST policies. Thus, it is reasonable that financial institutions and standardization bodies such as ISO/TC68 also advance the transition to new cryptographic algorithms based on NIST policies when dealing with Year 2010 issues.[17]

Such responses are also important for maintaining the confidence in the international standards developed by ISO/TC68 from the viewpoint of the security. Even if the transition to stronger algorithms would fail in 2010, it is unlikely that the security of IT systems adopting the current algorithms will simply collapse. However, if financial institutions would continue to use cryptographic algorithms which have already lost the guarantee for the security, the financial institutions would inevitably lose confidence in the security of their IT systems. More seriously, in the sense that such financial institutions fail to respond to Year 2010 issues in a timely manner despite the release of the NIST transition schedule in 2005, the reputation of the corresponding financial institutions would be damaged due to lack of their awareness regarding importance of information security measures.

Responses to Year 2010 issues target all cryptographic algorithms considered current mainstreams. The cryptographic algorithms in question are widely used in the financial sector, and the current situation differs from the past transitions as in the case of DES or triple DES. Further, in addition to growing numbers of financial institutions using cryptographic algorithms, a scope of financial services using cryptographic algorithms is also expanding, including interbank networks, IC cash cards, and Internet banking services. A transition to new cryptographic algorithms should be undertaken appropriately and timely in these diverse IT systems. Inevitably, the number of discussion points to be examined by financial institutions seems to be larger in Year 2010 issues than in the past situations. Therefore, it may take more time to successfully complete the transition in this time. These considerations suggest that responses to Year

---

[17] NESSIE and CRYPTREC evaluations also emphasize the results of NIST evaluations of cryptographic algorithms. Evaluations of AES by NESSIE highlight that NIST has performed detailed evaluations. The e-Government Recommended Ciphers List released by CRYPTREC [45] in February 21 2006, includes three-key triple DES, under the condition that it is specified as SP 800-67.

2010 issues should be initiated as soon as possible.

Given these points, we need to be aware of the importance of responses to Year 2010 issues, and begin to examine how we can best address these issues.

5. Addressing Year 2010 issues

Discussing how to address Year 2010 issues, the financial institutions need to consider which cryptographic algorithms to be selected and how IT systems will be updated to adopt the new algorithms. In this section, we will clarify major points to be discussed.

(1) Cryptographic algorithms specified, approved, and recommended by various organizations and projects

One possible course of action in selecting cryptographic algorithms for long-term use after 2010 is to select ones approved or recommended by NIST. However, in contrast to the time at which DES or triple DES was adopted, several alternative algorithms are now recommended by third party organizations other than NIST, comprised of cryptographers and other experts. For instance, CRYPTREC has developed the e-Government reccommended ciphers list, and NESSIE has also published the security evaluation results for several cryptographic algorithms. ISO/IEC JTC1/SC27, which is responsible for standardizing information security techniques for general purposes, has already developed ISO/IEC 18033 which specifies the cryptographic algorithms used to protect confidentiality. This standard was examined with reference to the evaluation results by NIST, CRYPTREC, and NESSIE. Therefore, ISO/IEC 18033 includes the cryptographic algorithms specified or recommended by these organizations and projects. In making final decisions concerning cryptographic algorithms, we may select algorithms suitable for various requirements not only from the algorithms approved or recommended by NIST but also those specified or recommended by the other organizations and projects.

In the previous sections, the cryptographic algorithms have been described at the very high level such as RSA in order to make it easier for the reader to understand. In this section, the algorithms specified or recommended in the organizations and projects will be expressed in the more detailed description in order to clarify the specifications and recommendations.

A. NIST policies

As indicated in SP 800-57, NIST recommends the algorithms and the key lengths listed below assuming use up to 2030 (See Table 4 (2)). We refer to [46] for hash functions.

- Symmetric ciphers: AES or three-key triple DES
- Asymmetric ciphers based on the factoring problem: With the minimum key

Table 5 (1) Cryptographic algorithms specified in ISO/IEC 18033

Classification		Cryptographic algorithms
Asymmetric ciphers (Part 2)		• Based on the factoring problem: RSA-KEM, RSA-OAEP, and HIME(R) • Based on DLP: ACE-KEM • Based on ECDLP: PSEC-KEM and ECIES-KEM
Symmetric ciphers	Block ciphers (Part 3)	• 64-bit block ciphers: CAST-128, MISTY1, and triple DES • 128-bit block ciphers: AES, Camellia, and SEED
	Stream ciphers (Part 4)	• MUGI • SNOW 2.0

length of 2048 bits

- Asymmetric ciphers based on DLP: With the minimum key length of 2048 bits. The size of order q of the subgroup over the finite field is 224 bits.
- Asymmetric ciphers based on ECDLP: With the minimum key length of 224 bits
- Hash functions: SHA-224, SHA-256, SHA-384, or SHA-512

For asymmetric ciphers, NIST approves RSA (quoting ANS X 9.31), DSA, and ECDSA in FIPS 186-2 (Digital Signature Standard)[18]. For the key agreement schemes, NIST recommends DH, MQV, and methods that implement each of these on elliptic curves in SP 800-56 [50]. NIST considers that the key lengths of DH and MQV are compatible with those for asymmetric ciphers based on the discrete logarithm problem described in the same SP.

B. ISO/IEC 18033

ISO/IEC 18033 is the first international standard for cryptographic algorithms used to protect confidentiality in general purposes. ISO/IEC JTC1/SC27 started the standardization activities in April 2000. This international standard has the following structure: Part 2 specifies asymmetric ciphers, Part 3 specifies block ciphers, and Part 4 specifies stream ciphers [32, 33, 34]. The cryptographic algorithms specified in ISO/IEC 18033 will be referred by other international standards that are currently being developed in ISO/IEC JTC1/SC27. Therefore, it is likely that the algorithms specified in ISO/IEC 18033 will be used widely in many fields.

[18] The draft of FIPS 186-3 describes the RSA algorithms specified in both ANS X9.31 and PKCS#1 version 2.1 (that is, RSASSA-PKCS1-v1_5 and RSA-PSS). In terms of DSA and ECDSA, the algorithms are the same as that specified in FIPS 186-2. In DSA, the new approved combinations of key length are added.

Developing ISO/IEC 18033, ISO/IEC JTC1/SC27 mandated security and performance evaluation results conducted by public organizations such as CRYPTREC and NESSIE as the necessary condition for the candidate algorithms. As the results, the proposed algorithms were those approved by NIST or those recommended by CRYPTREC or NESSIE. The candidate algorithms were comprehensively examined with respect to security and implementation performance in SC27. Finally, each part of this standard has come to specify two or more cryptographic algorithms.

Table 5 (1) summarizes the cryptographic algorithms specified in ISO/IEC 18033. For asymmetric ciphers, ISO/IEC 18033-2 separately specifies the algorithms based on the factoring problem, those based on DLP, and those based on ECDLP. All of these algorithms provide the provable security. However, ISO/IEC 18033-2 does not specify key lengths. Therefore, users of this international standard must the appropriate key length by themselves.

Specifications for symmetric ciphers are given separately for 64-bit block ciphers and 128-bit block ciphers. For 64-bit block ciphers, ISO/IEC 18033-3 specifies CAST-128, MISTY1, and triple DES. In particular, ISO/IEC 18033-3 states that two options are available for triple DES (two-key and three-key triple DES). However, ISO/IEC 18033-3 describes that NIST will recommend two-key triple DES up to approximately 2009 but no later, and recommends three-key triple DES. These statements appear to indicate that ISO/IEC 18033-3 is prompting users to adopt three-key triple DES if they must use triple DES. For 128-bit block ciphers, ISO/IEC 18033-3 specifies AES, Camellia, and SEED with key lengths set to 128 bits or longer.

For stream ciphers[19], it specifies MUGI and SNOW 2.0 but not RC4, which is now widely used.

C. e-Government Recommended Ciphers List by CRYPTREC

CRYPTREC's final goal was to develop the e-Government recommended ciphers list. This list consists of cryptographic algorithms suitable for IT systems in Japanese e-Government. CRYPTREC was established in May 2000 primarily to evaluate the candidate cryptographic algorithms to be incorporated into the e-Government recommended ciphers list[20]. CRYPTREC announced an open solicitation for

[19] The General model for stream ciphers in ISO/IEC 18033-4 specifies two methods for implementing stream cipher output functions: a method that calculates the XOR between the key stream and the plaintext and a method using MULTI-S01.

[20] See http://www.cryptrec.jp for information on recent CRYPTREC activities.

Table 5 (2): e-Government Recommended Ciphers List [45]

Category of technique		Cryptographic algorithms
Public-key cryptographic techniques	Signature	DSA, ECDSA, RSASSA-PKCS1-v1_5, and RSA-PSS
	Confidentiality	RSA-OAEP and RSAES-PKCS1-v1_5 (*1)
	Key agreement	DH, ECDH, and PSEC-KEM (*2)
Symmetric-key cryptographic techniques	64-bit block ciphers (*3)	CIPHERUNICORN-E, Hierocrypt-L1, MISTY1, and three-key triple DES (*4)
	128-bit block ciphers	AES, Camellia, CIPHERUNICORN-A, Hierocrypt-3, and SC2000
	Stream ciphers	MUGI, MULTI-S01, 128-bit RC4 (*5)
Other techniques	Hash functions	RIPEMD-160 (*6), SHA-1 (*6), SHA-256, SHA-384, and SHA-512
	Pseudo-random number generators (*7)	• PRNG based on SHA-1 in ANSI X9.42-2001 Annex C.1 • PRNG based on SHA-1 for general purpose in FIPS 186-2 (+ change notice 1) Appendix 3.1 • PRNG based on SHA-1 for general purpose in FIPS 186-2 (+ change notice 1) revised Appendix 3.1

Notes:

(*1) This is permitted to be used for the time being because it was used in SSL3.0/TLS1.0.

(*2) This is permitted to be used only in the KEM (key encapsulation mechanism) – DEM (data encapsulation mechanism) construction.

(*3) When constructing a new system for e-Government, 128-bit block ciphers are preferable if possible.

(*4) Three-key triple DES is permitted to be used for the time being under the following conditions:

(1) It is specified as SP 800-67,

(2) It is positioned as the de facto standard.

(*5) It is assumed that 128-bit RC4 will be used only in SSL3.0/TLS (1.0 or later). If any other cipher listed above is available, it should be used instead.

(*6) If a longer hash value is available when constructing a new system for e-Government, it is preferable to select a 256-bit (or more) hash function. However, this does not apply to the case where the hash function is designated to be used in the public-key cryptographic specification.

(*7) Since pseudo-random number generators do not require interoperability due to their usage characteristics, no problems will occur from the use of a cryptographically secure pseudo-random number generating algorithm. These algorithms are listed as exmples.

cryptographic algorithms required for e-Government, then evaluated the submitted algorithms based on reports of security evaluations entrusted to leading cryptographers within and outside Japan and papers presented at academic conferences. CRYPTREC published the e-Government recommended ciphers list in February 2003 (Table 5 (2), [45]).

Reorganized in fiscal 2003, CRYPTREC is now pursuing its activities under a new administrative umbrella. Namely, the Cryptographic Technique Monitoring Committee and the Cryptographic Technique Investigation Working Group monitor the emergence of security problems in cryptographic algorithms incorporated into the e-Government recommended ciphers list. The Cryptographic Module Committee performs researches for implementing and evaluating secure cryptographic modules.

In determining cryptographic algorithms to be evaluated, CRYPTREC included the

algorithms already deployed in a wide range of fields (so called *de facto* standards), as well as the submitted ones. CRYPTREC evaluated these algorithms and developed the e-Government Recommended Ciphers list considering the following points: (1) select several algorithms with sufficient security for the use in the e-Government system (security guaranteed roughly 10 years), and (2) select at least one algorithm pre-incorporated or likely to be incorporated in commercial puroducts for each category.

According to CRYPTREC Report 2002 [36], recommended key lengths for asymmetric ciphers as of 2002 are 1024 bits or longer for RSA, DSA and DH, and 160 bits or longer for ECDSA. The digital signature schemes for RSA include RSASSA-PKCS1-v1_5 and RSA-PSS, which are incorporated into PKCS #1 version 2.1. The encryption schemes for RSA include RSAES-PKCS1-v1_5 and RSA-OAEP, which are also incorporated into PKCS #1 version 2.1. CRYPTREC Report 2002 does not describe the evaluation results on advantages and drawbacks of these schemes. In general, RSA-PSS and RSA-OAEP are considered to be preferable from the viewpoint of security due to the fact that they are provably secure. Here, PSEC-KEM is recommended under the following condition: “This is permitted to be used only in the KEM (key encapsulation mechanism)[21] - DEM (data encapsulation mechanism)[22] construction.”

With regard to symmetric ciphers, the e-Government recommended ciphers list includes the following sentence as a note: “When constructing a new system for e-Government, 128-bit block ciphers are preferable if possible.” Focusing on 64-bit block ciphers, we can notice that two-key triple DES is excluded from the list of selected 64-bit block ciphers. In addition, the following note for three-key triple DES is described: “Three-key triple DES is permitted to be used for the time being under the following conditions: (1) It is specified as SP 800-67, (2) It is positioned as the de facto standard.” Although RC4 with a key length of 128 bits is also recommended,

---

[21] KEM is a cryptographic mechanism based on asymmetric ciphers for the purpose of transmission of session keys used for symmetric ciphers. The KEM encryption process, which is performed by the sender, takes receiver’s public key and certain parameters as input to the encryption function and outputs a session key *K* and data *C* required for obtaining *K*. The sender transmits *C* to the receiver and secretly stores *K* at the same time. The decryption process, which is performed by the receiver, takes receiver's secret key and *C* as inputs to the decryption function and outputs *K*. As the result, *K* can be shared between the sender and the receiver. If the receiver cannot obtain the correct key *K*, data *C* is automatically destroyed.

[22] DEM is a cryptographic mechanism based on symmetric cryptographic techniques, which protects both confidentiality and integrity. DEM is provably secure under an assumption that a symmetric cipher adopted satisfies certain security characteristics. ISO/IEC 18033-2 specifies a mechanism that combines DEM with an asymmetric cipher implementing KEM (the KEM-DEM construction). In this case, the entire mechanism based on the KEM-DEM construction is also provably secure.

the following note is described: “It is assumed that 128-bit RC4 will be used only in SSL3.0/TLS (1.0 or later). If any other cipher listed above is available, it should be used instead.”

Although the list of hash functions includes SHA-1, the following note is given: “If a longer hash value is available when constructing a new system for e-Government, it is preferable to select a 256-bit (or more) hash function. However, this does not apply to the case where the hash function is designated to be used in the public-key cryptographic specification.” On this basis, CRYPTREC appears to recommend SHA-256, SHA-384, and SHA-512 as acceptable hash functions when new IT systems of e-Government are constructed in the future.

D. NESSIE-recommended ciphers

The European Union (EU) launched the NESSIE project in 2000 as part of the Fifth EU Information Society Technologies Program. The main objective of the NESSIE project was to evaluate candidate algorithms and produce a list of recommended algorithms (the NESSIE portfolio). The goal of the project was to help strengthen the position of European industry in cryptography, as well as that of European research capabilities. According to [56], it was intended that the results of the NESSIE projects would be used as input material to various standardization activities in order to build consensus for the standardization of cryptographic algorithms. In fact, the achievements discussed here as NESSIE-recommended ciphers were provided to the standardization activities in ISO/IEC and IETF. Cryptographers from European universities and security companies took the lead in NESSIE evaluations, comprehensively assessing security, implementation performance, and handling of intellectual property rights, and finally selecting the recommended algorithms listed in Table 5 (3). NESSIE itself was dissolved after the compilation of the final report in March 2003.

Table 5 (3) NESSIE-recommended ciphers

Classification		Names of cryptographic algorithms
Asym-metric ciphers	Signature	RSA-PSS (primary recommendation)(*1), ECDSA (secondary recommendation)(*2), and SFLASH (for special applications)
	Confidentiality	PSEC-KEM (primary recommendation)(*2), RSA-KEM (secondary recommendation)(*1), and ACE-KEM (for special applications)(*3)
	Authentication	GPS
Sym-metric ciphers	64-bit block ciphers	MISTY1
	128-bit block ciphers	AES、Camellia
	256-bit block ciphers	SHACAL-2
Other	Hash functions	SHA-256, SHA-384, and SHA-512, Whirlpool (*4)
	Message authentication code	UMAC、TTMAC、EMAC、HMAC

Notes:

(*1) Key lengths of 1536 bits or longer are recommended to ensure medium-term security (5 to 10 years).

(*2) Key lengths of 160 bits or longer are recommended to ensure medium-term security.

(*3) To ensure medium-term security, key lengths of 160 bits or longer are recommended when using elliptic curves and key lengths of 1536 bits or longer when using finite fields.

(*4) The size of the hash value for Whirlpool is 512 bits.

(Reference) This table provides a summary of the evaluation results in the NESSIE consortium [54] (Dated February 27, 2003)

NESSIE recommendations specify RSA-PSS (primary recommendation) and ECDSA (secondary recommendation) as the digital signature schemes. They specify PSEC-KEM (primary recommendation) and RSA-KEM (secondary recommendation) as the encryption schemes. The key lengths are specified in such a way to ensure an adequate security level over the medium term (5 to 10 years). Key lengths of 1536 bits or longer are recommended for RSA based schemes. Key lengths of 160 bits or longer are recommended for ECDSA and PSEC-KEM, elliptic curve cryptosystems. Addressing Year 2010 issues will require long-term security, rather than the medium term security assumed for NESSIE evaluations. Thus, in order to determine the key lengths for the long-term security on the basis of the NESSIE evaluation results, key lengths of at least greater than 1536 and 160 bits must be set for RSA based schemes and elliptic curve cryptosystems, respectively.

For symmetric ciphers, MISTY1 is recommended as a 64-bit block cipher, while AES and Camellia are recommended as 128-bit block ciphers. NESSIE does not recommend any stream ciphers including RC4.

Regarding hash functions, all of four recommended hash functions have hash values of 256 bits or longer, consistent with CRYPTREC evaluations of hash functions in e-Government Recommended Ciphers List.

E. Comparison of specified, approved, and recommended cryptographic

algorithms

In the previous sections, we introduced cryptographic algorithms approved or recommended by NIST, those specified in ISO/IEC 18033, and those reccommeded by CRYPTREC and NESSIE. Table 5 (4) provides an overview of these algorithms.

At first, let us compare the asymmetric ciphers without biase. We see that RSA and ECDSA are the cryptographic algorithms approved or recommended by NIST, CRYPTREC, and NESSIE as a digital signature scheme. In terms of RSA, NIST recommends RSA specified in ANS X9.31 and PKCS#1 (version 1.5 and higher) with a key of 2048 bits or longer. CRYPTREC recommends both RSA-PSS and RSASSA-PKCS1-v_5 with a key of 1024 bits or longer, while NESSIE recommends only RSA-PSS with a key of 1536 bits or longer. In terms of ECDSA, CRYPTREC and NESSIE recommend ECDSA with a key of 160 bits or longer, and NIST recommends 224 bits or longer. DSA is approved and recommended by NIST and CRYPTREC, respectively. However, the recommended key lengths differ.

With regard to asymmetric ciphers for assuring confidentiality, ISO/IEC 18033-2 specifies six types of cryptographic algorithms. Among them, the e-Government Recommended Ciphers List includes RSA-OAEP, while NESSIE-recommended ciphers include PSEC-KEM, RSA-KEM, and ACE-KEM. Based on the evaluation results from CRYPTREC and NESSIE, no cryptographic algorithms are recommended commonly by both organizations. Note that NESSIE recommends only KEM schemes which have the key agreement mechanism. Recommended key lengths for RSA also differ.

In terms of key agreement schemes, if we regard PSEC-KEM as a key agreement scheme, PSEC-KEM can be found among both the NESSIE-recommended ciphers and the e-Government recommended ciphers list. A 160-bit key length for PSEC-KEM is also the same for both the evaluation results. DH is also recommended by both NIST and CRYPTREC.

Among symmetric 64-bit block ciphers, no cryptographic algorithms are simultaneously specified in ISO/IEC 18033-3, approved by NIST, and recommended by CRYPTREC and NESSIE. However, MISTY1 is specified in ISO/IEC 18033-3 and recommended by CRYPTREC and NESSIE. For 128-bit block ciphers, AES is specified in ISO/IEC 18033-3, approved by NIST and recommended by both CRYPTREC and NESSIE. Camellia is specified in ISO/IEC 18033-3 and recommended by both CRYPTREC and NESSIE. RC4 is recommended only by CRYPTREC under the certain conditions.

For hash functions, SHA-256, SHA-384, and SHA-512 are approved by NIST and recommended by both CRYPTREC and NESSIE.

Table 5 (4) Cryptographic algorithms specified, approved, or recommended by NIST, CRYPTREC, NESSIE and ISO

		NIST FIPSs and SPs (assuming use up to the end of 2030)	e-Government recommended ciphers list of CRYPTREC	NESSIE-recommended ciphers	ISO/IEC 18033
Asym-metric ciphers	Signature	• RSA specified in ANS X9.31 (2048 bits) • DSA (2048 bits) • ECDSA (224 bits)	• DSA (1024 bits) • ECDSA (160 bits) • RSASSA-PKCS1-v1_5 (1024 bits) • RSA-PSS (1024 bits)	• RSA-PSS (1536 bits) • ECDSA (160 bits) • SFLASH	(Specified in ISO/IEC 9796-2 and 14888-3)
	Confiden-tiality	(No recommendation)	• RSA-OAEP (1024 bits) • RSAES-PKCS1-v1_5 (1024 bits)	• PSEC-KEM (160 bits) • RSA-KEM (1536 bits) • ACE-KEM	• RSA-KEM • RSA-OAEP • HIME(R) • ACE-KEM • PSEC-KEM • ECIES-KEM
	Key agreement	• DH • MQV	• DH (1024 bits) • ECDH (160 bits) • PSEC-KEM (160 bits)	(No recommen-dation)	(Specified in ISO/IEC 11770-3)
Sym-metric ciphers	64-bit block ciphers	Three-key triple DES	• CIPHERUNICORN-E • Hierocrypt-L1 • MISTY1 • Three-key triple DES	MISTY1	• CAST-128 • MISTY1 • Triple DES (three-key recommended)
	128-bit block ciphers	AES	• AES • Camellia • CIPHERUNICORN-A • Hierocrypt-3 • SC2000	• AES • Camellia	• AES • Camellia • SEED
	Stream ciphers	(No recommendation)	• MUGI • MULTI-S01 • RC4 (128 bits)	(No recommen-dation)	• MUGI • SNOW 2.0
Hash functions		• SHA-224 • SHA-256 • SHA-384 • SHA-512	• RIPEMD-160 • SHA-1 • SHA-256 • SHA-384 • SHA-512	• SHA-256 • SHA-384 • SHA-512 • Whirlpool	(Specified in ISO/IEC 10118)

Note: The key lengths in the table are the minimum recommended values.

(2) Major points of discussion in selecting cryptographic algorithms

In this subsection, we will show several major points to be discussed in order to select appropriate cryptographic algorithms and key lengths.

[Discussion point 1] How to weigh and interpret evaluation results from the various organizations and projects

Of the evaluation results given by ISO/IEC 18033, FIPSs and SPs of NIST, CRYPTREC, and NESSIE, the financial institutions need to first consider which evaluation results are regarded as being the most suitable for their IT systems. In order to do so, it is important that the financial institutions remain aware of differences among the evaluation criteria applied by these organizations and projects. When evaluating cryptographic algorithms, NIST and NESSIE considered implementation performance as well as security; On the other hand, CRYPTREC considered not only security but also availability at the time when the e-Government recommended ciphers list has been published. Given these differences in evaluation criteria, it is necessary to determine which evaluation results for cryptographic algorithms should be mainly referred to. The financial institutions can focus on the algorithms which are specified, approved or recommended on the basis of the evaluation results they selected.

[Discussion point 2] Selecting a key length if different key lengths are recommended, specified or approved by the organizations or projects for the same cryptographic algorithm

After selecting specific cryptographic algorithms, the financial institutions have to determine their key lengths. It is important to consider differences of the approaches taken by the organizations or projects. In the case of the e-Government recommended ciphers list, CRYPTREC selected cryptographic algorithms based on their availability for e-Government systems at the time when the list was released. For this reason, the recommended minimum key lengths tend to be shorter than those recommended by other organizations or projects. On the other hand, NIST selects the minimum key lenghs for the use of the algorithms in various time periods such as throughout and after 2030, as well as throughout 2010.

In principle, it is desirable to adopt the largest value among the recommended minimum key lengths from the viewpoint of security. However, in cases requiring the interoperability with other financial IT systems, the key lengths in question may be set to the same value as those of the systems. Additionally, it may not be possible to physically implement a desirable key length due to various conditions such as smart card performance. When selecting the optimal key length, the financial institutions need to consider all of these diverse factors, which depend on features of an application to be discussed.

[Discussion point 3] Choosing a 64-bit or 128-bit block length for symmetric ciphers

For symmetric ciphers, ISO/IEC 18033-3, NIST FIPSs/SPs, the e-Government

recommended ciphers list, and the NESSIE-recommended ciphers include both 64-bit and 128-bit block ciphers. However, NIST FIPSs and the NESSIE-recommended ciphers exclude triple DES, which is the representative 64-bit block cipher. The e-Government recommended ciphers list suggests a preference for 128-bit block ciphers if they are available in products. Taking these into considerations, the financial institutions should focus on 128-bit block ciphers at first. If the selection is based on ISO/IEC 18033-3, the financial institutions should consider AES, Camellia, and SEED as primary candidates.

Stream ciphers may achieve higher performance in encryption and decryption operations than existing block ciphers. However, these cases are considered relatively rare. Although ISO/IEC 18033-4 specifies MUGI and SNOW 2.0, neither NIST nor NESSIE approves or recommends stream ciphers. Currently, methods for the security evaluation for stream ciphers tend to be less established than for block ciphers. Where block ciphers (for which a relatively large number of security evaluations have been conducted and published) can be used, there is little pressing need to adopt stream ciphers in place of block ciphers.

[Discussion point 4] Determining the size of the hash value for hash functions

SHA-224, SHA-256, SHA-384 and SHA-512 (hereafter referred to as SHA-2) are approved by NIST, while SHA-256, SHA-384, and SHA-512 are recommended by CRYPTREC and NESSIE. These must be considered to arrive at the optimal choice. Basically, it is necessary to address whether the size of the hash value is compatible with the specifications of the corresponding IT systems.

Nevertheless, the financial institutions should keep in mind that NIST itself is now considering whether or not it is appropriate to move to SHA-2. As indicated in comments on the successful attack to SHA-0, NIST was planning a transition to SHA-2 for its higher levels of security, while assuming that SHA-1 would remain secure for the foreseeable future. However, contrary to all expectations, a successful attack has also been launched against SHA-1. This fact raised questions as to the security of SHA-2, since its design philosophy follows the basis of that of SHA-1. It is believed that NIST is now conducting various studies, including those exploring the need to develop a new hash function replacing or coexisting with SHA-2.

Since it is all but certain that SHA-1 will be removed from the US Government standards by 2010, the transition to SHA-2 shall appear to be necessary, although it may be temporary. However, for the medium or long term, the financial institutions must keep in mind the possibility of the addition of or the transition to new hash functions other than SHA-2.

[Discussion point 5] Considering performance aspects such as processing time in

encryption and decryption operations when selecting the cryptographic algorithms

Let us suppose that cryptographic algorithms are needed to run on processors with relatively low computational power such as smart cards in a certain system and that its user requires to balance security with processing time in encryption and decryption operations. Under such a circumstance, of those that can provide approximately the same security levels, the user would select cryptographic algorithms that meet specific requirements on the implementation perforamnce. Thus, it is necessary to determine in advance the implementation requirements by taking into consideration the objective system since the content of such requirements generally depends on features of the application.

The following may be also one discussion point when discussing how to implement asymmetric ciphers.

[Discussion point 6] Determining whether or not to adopt KEM-DEM construction for an asymmetric cipher

With regard to asymmetric ciphers, ISO/IEC 18033-2, the e-Government recommended ciphers list, and the NESSIE-recommended ciphers include KEM as the recommended algorithms for the key agreement purpose. Especially, ISO/IEC 18033-2 specifies the use of KEM in the form of “KEM-DEM construction,” which combines KEM with DEM. This is the case of those included in the e-Government recommended ciphers list by CRYPTREC.

Traditionally, cryptographic algorithms for key agreement have been discussed differently from those for confidentiality or integrity. However, it is currently possible to design and develop hybrid schemes which can be used not only for key agreement but also for confidentiality or integrity by adopting the KEM-DEM construction. Since the KEM-DEM construction has been recommended in ISO/IEC 18033-2, the NESSIE-recommended ciphers and the e-Government recommended ciphers list, the construction may be deployed in various scenes in the coming years. Therefore, it may be beneficial for the financial institutions to consider the possibility of adopting the KEM-DEM construction as an alternative.

In applications involving asymmetric ciphers applied to ensure confidentiality, it may be relatively easy to introduce the KEM construction, since asymmetric ciphers are generally used to distribute session keys for symmetric ciphers. However, implementing the ciphers in the KEM-DEM construction may require additional system modifications. This is because the approach differs from the conventional use of asymmetric ciphers and symmetric ciphers. The deployment of the KEM-DEM construction may require the system modifications to the other financial institutions. All of these points are important

and are needed to be confirmed when the financial institutions consider to adopt the KEM-DEM construction.

(3) Other points requiring attention

The three following points require attention in revising specifications for IT systems associated with the transitions to new cryptographic algorithms.

[Points requiring attention 1] When changing the specifications of cryptographic algorithms, ensure that such changes will not impair the interoperability for the IT system in question and other related systems.

In many cases, financial institution's IT systems are linked to other systems. For instance, if a bank changes the cryptographic algorithm used in its ATM network system and updates its IC cash card and ATM specifications, the IC cash cards may no longer work in ATMs operated by other banks, or the ATMs of the bank in question may not accept IC cash cards issued by the other banks.

To prevent such problems, the financial institutions must discuss impacts of modifying specifications of their IT systems on other related systems in the transition of cryptographic algorithms. In some cases, it may be necessary to consider separating the cryptographic algorithms in different IT systems. Additionally, with respect to the effects on other financial institutions, the financial institutions must consider in advance how to minimize potential damage – for instance, by coordinating the contents and timing of system modifications with other banks carefully.

[Points requiring attention 2] Revise configurations to ensure appropriate use of the new cryptographic algorithms and to prohibit use of the old algorithms.

Even if a financial institution introduces a new cryptographic algorithm at significant cost in response to Year 2010 issues, the measure will have little benefit if problems with the corresponding IT system lead to continued use of the old, insecure cryptographic algorithm. Such circumstances may arise with Internet banking systems. For instance, it is considered that inappropriate SSL server settings may permit to use not only symmetric ciphers with a 128-bit key, but also those with a 40-bit or 56-bit key. Eliminating the possibility of such problems requires configuring appropriate settings on the server to disable communications based on the old, insecure ciphers, in addition to moving to and enabling the new cryptographic algorithms. Such attention is required not just for Internet banking systems, but also for all information systems affected by the transition to the new cryptographic algorithms.

The following is also important although it may be security practice in general.

[Points requiring attention 3] Do not share cryptographic keys when circumstances require the use of two or more cryptographic algorithms, even if the algorithms are the same class of cryptographic techniques.

In certain cases, as in Points requiring attention 1, two or more separate cryptographic algorithms must be used to ensure the interoperability with other related systems. For instance, the following situation is assumed: while cryptographic algorithm is revised based on ISO/IEC 18033-3 to AES, Camellia, or SEED in the part of a system used within a bank, triple DES continues to be used in the part of the system open to other banks in order to maintain the interoperability.

In such cases, if the same cryptographic key is adopted both in the parts of the system, security levels depend on the weaker algorithm (in this case triple DES). As the results, an effect of moving to a more secure algorithm is lost. Even if cryptographic techniques are the same, sharing the keys raises significant security issues and shall be avoided.

(4) Medium-term issues

In Year 2010 issues, the deadline by which the issues must be addressed is quite clear because NIST has already disclosed a point of time by which the security guarantees will no longer hold. Although the likelihood may remain extremely low, even cryptographic algorithms that have been evaluated as providing adequate security may find vulnerable to unanticipated advances in cryptanalytic techniques. Users of cryptographic algorithms must remain aware of the possibility of sudden compromises. It is desirable not only for each financial institution but also for the financial industry as a whole to establish systems for dealing with future compromise of cryptographic algorithms appropriately by making use of the discussion on Year 2010 issues.

With respect to the establishment of such systems, the following two issues should be addressed before all others:

[Medium-term issue 1] Establish systems to keep current with the security evaluations of cryptographic algorithms and actions of other organizations and projects such as NIST, CRYPTREC, NESSIE and ISO.

When it remains unclear whether security guarantees for cryptographic algorithms will be lost, the financial institutions must decide on their own when they should move to new cryptographic algorithms. In order to aid such decision, it is necessary to monitor the security evaluation results for cryptographic algorithms already published and

continuously examine prospects for security based on information released by the evaluation organizations such as NIST. The security evaluation of cryptographic algorithms requires profound expertise. It is not easy for the financial institutions to establish such systems within a short timeframe. Nevertheless, since the financial institutions have significant responsibility for assuring the security of their IT systems which have a highly public aspect, the financial institutions are required to address such issues steadily and proactively.

The financial institutions would have to perform their own security evaluations if they adopted cryptographic algorithms which were not approved or recommended by the third parties such as ISO, NIST, CRYPTREC and NESSIE.

In the future, the financial institutions may also begin using services such as digital time stamps. In such a case, the financial institutions would be required to pay attention to the cryptographic algorithms used for these services. Digital time stamps are one possible means for ensuring integrity and other characteristics of digital data in the medium to long term, and one presumes that the cryptographic algorithms used in association with time stamps will provide medium- to long-term security. In this area as well, the financial institutions should establish systems for performing their own security evaluations of cryptographic algorithms.

[Medium-term issue 2] Study how to design and develop IT systems which can smoothly accommodate changes in cryptographic algorithms and in key lengths.

One way of smoothly dealing with the threat of compromise of cryptographic algorithms is to design and develop IT systems that can easily accommodate changes in cryptographic algorithms and key lengths. Examples include the use of cryptographic modules that allow easy replacement and/or communication formats that allow changes in key lengths and ciphertext sizes.

To prepare a ready response to compromise of cryptographic algorithms, one may also implement in advance two or more cryptographic algorithms with different security characteristics such that it is easy to switch one algorithm to another one. As the result, even if one of the algorithms is compromised, the other one will continue to provide adequate security without any losses due to the migration of the algorithms, helping to ensure the security level of the IT system as a whole.

Other measures to anticipate compromise of cryptographic algorithms include development of IT systems based on unconditional security, rather than computational security. The unconditional security is a security characteristic that, as long as an attacker cannot obtain a certain amount of information, the security level can be maintained independently on the computational power of the attacker. It may be

beneficial to follow the research results of this technique. In addition, research results of various cryptographic techniques based on quantum mechanics may be also beneficial.

6. Summary

Year 2010 issues pose potential significant issues for financial institutions using the conventional cryptographic algorithms such as two-key triple DES, 1024-bit RSA, and SHA-1. The financial industry has traditionally relied on cryptographic algorithms guaranteed by NIST (FIPS approval) and has made use of such guarantees as important benchmarks in selecting cryptographic algorithms. By selecting cryptographic algorithms based on NIST decisions, financial institutions have been able to justify their selection of cryptographic algorithms to their customers and other related parties in terms of security and reliability. Taking into account these experiences to date, financial institutions should understand the background of NIST decisions and examine appropriate response to Year 2010 issues in the future.

Compared to the late 1970s, when DES proliferated in private sectors after NIST selection of DES as a FIPS-approved cipher, a wide range of cryptographic algorithms and options is now available. Financial institutions can select from cryptographic algorithms objectively recommended by various cryptographers and organizations. For instance, we can refer cryptographic algorithms specified in ISO/IEC 18033, the CRYPTREC e-Government Recommended Ciphers List, and NESSIE-recommended ciphers are included in addition to NIST's FIPS-approved ciphers. The cryptographic algorithms each differ in terms of security properties, algorithm structures, and implementation performance. It is necessary for financial institutions to understand these characteristics when considering which cryptographic algorithms should be adopted.

When selecting cryptographic algorithms, financial institutions should also consider the compatibility of the algorithms with their current applications. They should consider how to manage such a transition to minimize disruptions or damage to the existing systems and to customer convenience. In addition, the financial institutions should take care to prevent continuing use of weak algorithms used before the transition. Year 2010 issues can also be understood as an opportunity to consider appropriate measures to prevent and respond to compromise of cryptographic algorithms. Ideally, studies should be undertaken to anticipate potential breaches of cryptographic algorithms. Systems to follow security evaluation results for cryptographic algorithms and the actions of NIST and other evaluation organizations and projects will be vital.

As shown in the previous section, there are many points to be discussed with regard to Year 2010 issues. There is relatively little time before 2010 for such work, and it is expected that the financial institutions rapidly respond to Year 2010 issues in the coming years.

References

[1] Adleman, Leonard M., “The function field sieve,” *Algorithmic Number Theory*, LNCS 887, Springer-Verlag, pp. 108-121, 1994.

[2] American National Standards Institute, *X9.52: Triple Data Encryption Algorithm Modes of Operation*, 1998.

[3] Blake-Wilson, Simon, Magnus Nystrom, David Hopwood, Jan Mikkelsen and Tim Wright, *RFC 3546: Transport Layer Security (TLS) Extensions*, 2003.

[4] -------, -------, -------, ------- and -------, *IETF Internet Draft: Transport Layer Security (TLS) Extensions*, 2005.

[5] Brent, Richard, “Recent progress and prospects for integer factorization algorithms,” *Proceedings of COCOON 2000*, LNCS 1858, Springer-Verlag, pp. 3-20, 2000.

[6] Certicom Research, *SEC 1: Elliptic Curve Cryptography, Version 1.0*, 2000.

[7] Coppersmith, D., “Fast evaluation of logarithms in fields of characteristic two,” *IEEE Transaction on Information Theory*, IT-30 (4), pp. 587-594, 1984.

[8] Dierks, Tim, and Christopher Allen, *RFC 2246: The TLS Protocol Version 1.0*, 1999.

[9] --------, and Eric Rescorla, *IETF Internet Draft: The TLS Protocol, Version 1.1*, 2005.

[10] ElGamal, Taher, “A subexponential-time algorithm for computing discrete logarithms over $GF(p^2)$,” *IEEE Transactions on Information Theory* 31, pp. 473-481, 1985.

[11] EMVco, *EMV Integrated Circuit Card Specifications for Payments Systems Version 4.1, Book 2: Security and Key Management*, 2004.

[12] -------, *EMV Issuer and Application Security Guidelines, Version 1.3*, 2005.

[13] European Committee for Standardization and Information Society Standardization System (CEN/ISSS), *Pr CWA 14174-3: Financial transactional IC card reader (FINREAD) - Part 3: Security requirements*, 2002.

[14] Factor World, *General Purpose Factoring Records.* (http://www.crypto-world.com/FactorRecords.html)

[15] Fluhrer, Scott, Itsik Mantin, and Adi Shamir, “Attacks On RC4 and WEP,” *CryptoBytes* 5 (2), 2002. (http://www.rsasecurity.com/rsalabs/cryptobytes/index.html)

[16] Franke, Jens, Thorsten Kleinjung, Christof Paar, Jan Pelzl, Christine Priplata, and Colin Stahlke, “SHARK: A Realizable Special Hardware Sieving Device for Factoring 1024-Bit Integers,” *Proceedings of CHES 2005*, LNCS 3659, Springer-Verlag, pp. 119-130, 2005.

[17] Frey, G., and H.-G. Rück, “A Remark Concerning m-divisibility and the Discrete Logarithm in the Divisor Class Group of Curve,” *Mathematics of Computation*, 62 (206), pp.

865-874, 1994.

[18] Geiselmann, W., A. Shamir, R. Steinwandt, and E. Tromer, "Scalable Hardware for Sparse Systems of Linear Equations with Applications to Integer Factorization," *Proceedings of CHES 2005*, LNCS 3659, Springer-Verlag, pp. 131-146, 2005.

[19] Gordon, D., "Discrete Logarithms in *GF*(*p*) Using the Number Field Sieve," *SIAM Journal on Discrete Mathematics* 6, pp. 124-138, 1993.

[20] International Organization for Standardization (ISO), *ISO 9564-1: Banking -- Personal Identification -- Number management and security – Part 1: Basic principles and requirements for online PIN handling in ATM and POS systems*, 2002.

[21] -------, *ISO 9564-2: Banking -- Personal Identification -- Number management and security -- Part 2: Approved algorithms for PIN encipherment*, 2005.

[22] -------, *ISO 11568-2: Banking -- Key management (retail) -- Part 2: Key management techniques for symmetric ciphers*, 2005.

[23] -------, *ISO 11568-4: Banking -- Key management (retail) -- Part 4: Key management techniques using public key cryptography*, 1998.

[24] -------, *ISO 13491-2: Banking -- Secure cryptographic devices (retail) -- Part 2: Security compliance checklists for devices used in financial transactions*, 2005.

[25] -------, *ISO/TR 13569: Banking and related financial services -- Information security guidelines, Amendment 1*, 1998.

[26] -------, *ISO/TR 17944: Banking -- Security and other financial services -- Framework for security in financial systems*, 2002.

[27] ------- and International Electrotechnical Commission (IEC), *ISO/IEC 10118-2: Information technology -- Security techniques -- Hash functions -- Part 2: Hash functions Using an n-bit Block Cipher Algorithm*, 2000.

[28] ------- and -------, *ISO/IEC 10118-3: Information technology -- Security techniques -- Hash functions -- Part 3: Dedicated Hash functions*, 2004.

[29] ------- and -------, *ISO/IEC 10118-4: Information technology -- Security techniques -- Hash functions -- Part 4: Hash functions Using Modular Arithmetic*, 1998.

[30] ------- and -------, *ISO/IEC 11770-3: Information technology -- Security techniques -- Key management -- Part 3: Mechanisms using asymmetric techniques*, 1999.

[31] ------- and -------, *ISO/IEC 14888-3: Information technology -- Security techniques -- Digital signatures with appendix -- Part 3: Certificate-based mechanisms*, 1998.

[32] ------- and -------, *ISO/IEC 18033-2: Information technology -- Security techniques -- Encryption algorithms -- Part 2: Asymmetric ciphers*, 2005.

[33] ------- and -------, *ISO/IEC 18033-3: Information technology -- Security techniques -- Encryption algorithms -- Part 3: Block ciphers*, 2005.

[34] ------- and -------, *ISO/IEC 18033-4: Information technology -- Security techniques -- Encryption algorithms -- Part 4: Stream ciphers*, 2005.

[35] ------- and -------, *ISO/IEC 9796-2: Information technology -- Security techniques -- Digital signature schemes giving message recovery -- Part 2: Integer factorization based mechanisms*, 2002.

[36] Information-technology Promotion Agency, Japan (IPA) and Telecommunications Advancement Organization of Japan (TAO), *CRYPTREC Report 2002*, March 2003.

[37] Japan Bankers Association (JBA), *Standard Specifications of IC Cash Cards*, 2006 (in Japanese)

[38] Kaliski, B., *TWIRL and RSA Key Size*, 2003. (http://www.rsasecurity.com/rsalabs/node.asp?id=2004)

[39] Kusuda, K., and T. Matsumoto, "A Strength Evaluation of the Data Encryption Standard," *IMES Discussion Paper Series*, 97-E-5, Bank of Japan, 1997.

[40] Lenstra, A. K., and E. R. Verheul, "Selecting Cryptographic Key Size," *Journal of Cryptology*, 14 (4), pp.255-293, 2001.

[41] --------, X.Wang, and B. de Weger, "Colliding X.509 Certificates," Cryptology ePrint Archive, 2005. (http://eprint.iacr.org/2005/067)

[42] Lucks, S., and M. Daum, "The Story of Alice and her Boss: Hash Functions and the Blind Passenger Attack," *Presentation at Rump Sessions of Eurocrypt 2005*, 2005.

[43] Menezes, A. J., T. Okamoto, and S. A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pp. 80-89, 1991.

[44] Merkle, R., and M. Hellman, "On the Security of Multiple Encryption," *Communication of ACM*, 24 (7), pp.465-467, 1981.

[45] Ministry of Internal Affairs and Communication and Minstry of Economy, Trade and Industry, *e-Government recommended ciphers list*, Februay 20, 2003. (http://www.cryptrec.jp/images/cryptrec_01.pdf)

[46] National Institute of Standards and Technology (NIST), *Recommendation on Key Management, SP800-57*, 2005. (http://csrc.nist.gov/publications/nistpubs/800-57/SP800-57-Part1.pdf)

[47] -------, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification, SP800-78*, 2005. (http://csrc.nist.gov/publications/nistpubs/800-78/sp800-78-final.pdf)

[48] -------, *Draft Federal Information Processing Standard (FIPS) 186-3 - Digital Signature Standard (DSS)*, 2006.

[49] -------, *Personal Identity Verification of Federal Employees and Contractors, FIPS 201*, 2005. (http://csrc.nist.gov/publications/fips/fips201/FIPS-201-022505.pdf)

[50] -------, *Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, SP 800-56*, Draft, 2005.

[51] -------, *NIST Brief Comments on Recent Cryptanalytic Attacks on Secure Hashing*

Functions and the Continued Security Provided by SHA-1, 2004. (http://csrc.nist.gov/NIST%20Brief%20Comments%20on%20Hash%20Standards%208-25-2004.pdf)

[52] -------, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, SP800-67*, 2004.

[53] -------, *NIST Brief Comments on Recent Cryptanalytic Attacks on SHA-1*, 2005. (http://csrc.nist.gov/news-highlights/NIST-Brief-Comments-on-SHA1-attack.pdf)

[54] New European Schemes for Signatures, Integrity, and Encryption (NESSIE) consortium, *Portfolio of recommended cryptographic primitives*, 2003. (https://www.cosic.esat.kuleuven.be/nessie/ deliverables/decision-final.pdf)

[55] Pomerance, C., "Fast, Rigorous Factorization and Discrete Logarithm Algorithms," *Discrete Algorithms and Complexity*, edited by D. S. Johnson, T. Nishizeki, A. Nozaki, and H. S. Wilf, Academic Press, pp. 119-143, 1987.

[56] Preneel, B., A. Biryukov, C. De Cannière, S. B. Örs, E. Oswald, B. Van Rompay, L. Granboulan, E. Dottax, G. Martinet, S. Murphy, A. Dent, R. Shipsey, C. Swart, J. White, M. Dichtl, S. Pyka, M. Schafheutle, P. Serf, E. Biham, E. Barkan, Y. Braziler, O. Dunkelman, V. Furman, D. Kenigsberg, J. Stolin, J.-J. Quisquater, M. Ciet, F. Sica, H. Raddum, L. Knudsen, and M. Parker, *Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption*, Version 0.15 (beta), 2004.

[57] Sato, T., and K. Araki, "Fermat Quotients and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves," *Commentarii Mathematici Universitatis Sancti Pauli* 47 (1), pp. 81-92, 1998.

[58] Schirokauer, O., "Discrete Logarithms and Local Units," *Journal of Philosophical Transactions of the Royal Society of London*, Series A, Vol. 345, pp. 409-423, 1993.

[59] -------, D. Weber, and T. Denny, "Discrete Logarithms: The Effectiveness of the Index Calculus Method," *Algorithmic Number Theory*, LNCS 1122, Springer-Verlag, pp. 335-361, 1996.

[60] SWIFT, "Secure Card Reader Upgrade Coming Up," *SWIFT Bulletin*, No.13, pp.1-2, 2000.

[61] Taniguchi, F., K. Ohta, and M. Ohkubo, "Recent standardization trend around Triple DES," *Monetary and Economic Studies*, 18 (S-1), Institute for Monetary and Economic Studies, Bank of Japan, pp. 29-50, 1999.

[62] Thomé, E., *Discrete Logarithms in GF(2^607)*, 2002. (http://www.lix.polytechnique.fr/Labo/Emmanuel.Thome/announcement/announcement.html)

[63] Wang, X., "Cryptanalysis for Hash Functions and Some Potential Dangers," *RSA Conference 2006 Cryptographer's Track*, 2006.

[64] -------, A. Yao, and F. Yao, "New Collision Search for SHA-1," *Presentation of Rump*

Session of CRYPTO 2005, 2005. (http://www.iacr.org/conferences/crypto2005/r/2.pdf)

[65] -------, Y. L. Yin, and H. Yu, "Finding Collisions in the Full SHA-1," *Advances in Cryptology - CRYPTO 2005*, LNCS 3621, Springer-Verlag, pp.17-36, 2005. (http://www.infosec.sdu.edu.cn/paper/sha1-crypto-auth-new-2-yao.pdf)

[66] Yoshida, M., K. Kobara, and H. Imai, "Verification of WEP implementation in latest products," *Proceedings of 28th International Symposium on Information Theory and Applications (ISITA'05)*, 2005.