PIATS: A Partially Sanitizable Signature Scheme

Tetsuya Izu, Nobuyuki Kanaya, Masahiko Takenaka, and Takashi Yoshioka

Fujitsu Laboratories Ltd.,
4-1-1, Kamikodanaka, Nakahara-ku, Kawasaki, 211-8588, Japan
{izu, kanaya, takenaka, yoshioka}@labs.fujitsu.com

Abstract. In e-government or e-tax payment systems, appropriate alterations on digitally signed documents are required to hide personal information, namely privacy. Standard digital signature schemes do not allow such alternations on the signed documents since there is no means to distinguish appropriate alternations from inappropriate forgeries. The *sanitizable signature scheme* is a possible solution for such systems in which sanitizings of partial information are possible, after a signature is signed on the original (unsanitized) document. However, in previously proposed schemes, since sanitizers are anonymous, verifiers cannot identify sanitizers, and thus dishonest sanitizings are possible. This paper proposes a new sanitizable signature scheme "PIATS" in which partial information can be sanitized. Moreover, verifiers can identify sanitizers and thus dishonest sanitizings are eliminated.

Keywords: Sanitizable signature scheme, partial integrity, privacy.

1 Introduction

To governmental, municipal or military offices, there is a strong demand to disclose documents they hold or held. In fact, many countries have disclosure laws for these organizations. However, such disclosed documents should exclude personal or national secret information because of privacy or diplomatic reasons. In old days, paper documents were blacked-out by physical maskings to hide information. Unfortunately, we have no analogous system for electronic documents. For example, the New York Times website exposed CIA agents with carelessness, since they sanitized the electronic document by hand [Wir02]. Other example is an exposure of the Carnivore review team by the Justice Department of USA [Wir00]. Thus a systematic sanitizing method for electronic documents are required in this internet era. On the other hand, signatures are very common technology to assure the integrity of the document, since it detects inappropriate forgeries on original documents. However, current signature schemes can not distinguish appropriate alternations, namely sanitizations as mentioned above, and adversaries' inappropriate forgeries. Thus we require a new document managing system which establish the privacy of some part of documents (by sanitizations) and the integrity of other parts (by enhanced signature technology).

S. Qing et al. (Eds.): ICICS 2005, LNCS 3783, pp. 72–83, 2005.

The *sanitizable signature scheme* (or the *content extractable signature scheme*) is a possible solution in which partial information are sanitizable even after a signature is signed on the original document [SBZ01, MSI+03, MIM+05, ACM+05]. In these sanitizable signature schemes, appropriate alternations (sanitizings) and inappropriate alternations (forgeries) are distinctly treated, namely, all sanitizations are allowed but any forgeries are not. Thus these schemes guarantee the integrity of the original document with hiding the privacy. However, previously proposed sanitizable signature schemes required rather severe limitations or had serious security problems. In SUMI-1, a sanitization is allowed only once [MSI+03]. In CES schemes and SUMI-2,3,4, multiple sanitizations are allowed [SBZ01, MSI+03], however, since sanitizers are anonymous, verifiers cannot identify sanitizers. More worse, dishonest sanitizations are possible in SUMI-4 [MIM+05]. In a recent scheme SUMI-5 [MIM+05], such dishonestly sanitizations are avoided, but limitations are somewhat strengthened. In fact, SUMI-5 assumes a situation where sanitizers can control the disclosing criteria of the following sanitizers.

This paper proposes a new partially sanitizable signature scheme "PIATS" (Partial Information Assuring Technology for Signature) in which dishonestly sanitizations on any part of closed information are detected. Moreover, for sanitized information, verifiers can identify not only which part is sanitized but also who sanitized from a signer and sanitizers. Since any provably secure digital signature schemes such as RSA-PSS [PKCS] can be combined with the proposed scheme, we can establish an electronic document management system which assures the integrity of the document with hiding personal information.

The rest of the paper is organized as follows: section 2 describes previously proposed sanitizable signature schemes CES families and SUMI families. Then, section 3 proposes our sanitizable signature scheme with some discussions. A comparison of mentioned schemes are in section 4.

2 Preliminaries

In this section, we briefly introduce previously proposed sanitizable signature schemes CES families [SBZ01] and SUMI families [MSI+03, MIM+05].

2.1 Notations

In this paper, we assume that an original document to be digitally signed is given as an n-block data $\{m_i\}_{1 \leq i \leq n}$. Here length of each block can be distinct. For example, m_i can be minimal components in XML document. A value r is a (pseudo) random value generated by an appropriate generator. A function Hash$(\cdot)$ is an arbitrary secure hash function such as SHA-256 [1]. Functions Sign$(\cdot)$/Verify$(\cdot)$ are signing/verifying functions of a non-sanitizable provably secure signature scheme such as RSA-PSS [PKCS].

[1] As in [SBZ01], all hash functions in this paper can be replaced by preimage resistant and 2nd preimage resistant functions. However, for simplicity, we denote just "hash functions".

2.2 Three-Party Model

In the followings, we use a *three-party model* of sanitizable signature schemes as in previous schemes [MSI+03, MIM+05, SBZ01]. In this model, three parites, *signers*, *sanitizers* and *verifiers* are considered as players [2].

Signer assures the integrity of an original document by digitally signing a signature. The signer does not know which part of the document will be sanitized in future when they sign.

Sanitizers determine which part of the document to be sanitized and actually sanitize the document, on input a signed document and a signature from the signer (and a sanitized document if the sanitizer is not the first sanitizer). Here, we assume that sanitizers cannot create a new document, and sanitizers cannot change the original document nor signature.

Verifiers confirm the integrity of the document by verifying the original signature, and confirm whether it was signed by an appropriate signer and was sanitized by appropriate sanitizers.

2.3 CES

CES (Content Extracting Signature) is a family of sanitizable signature schemes proposed by Steinfeld-Bull-Zheng [SBZ01]. CES family has four schemes CES-CV, CES-HT, CES-RSAP, and CES-MERP. CES-CV is a main scheme and remaining are its variants. CES-CV and CES-HT can be combined with arbitrary signature schemes, while CES-RSAP and CES-MERP can be combined with only RSA-type signature schemes. Since CES-CV is very similar to SUMI-4 in the following section, we do not introduce CES schemes in detail here.

2.4 SUMI

SUMI is a family of sanitizable signature schemes proposed by Miyazaki et al. successively [MSI+03, MIM+05]. SUMI family has five schemes SUMI-1, SUMI-2, SUMI-3, SUMI-4, and SUMI-5 [3]. All of these schemes can be combined with arbitrary signature schemes.

SUMI-1, SUMI-2, SUMI-3: On an original n-block document, SUMI-1 generates signatures for all possible subsets of the document (namely, a signer generates 2^n signatures). A sanitizer determines disclosing blocks and publishes a corresponding subset and a signature. Thus a sanitization is allowed only once in SUMI-1. In addition, SUMI-1 is far from efficient since it requires 2^n signatures, which is exponential to the size of the original document.

SUMI-2 generates n signatures corresponding to n-blocks. A sanitizer determines disclosing blocks and publishes a corresponding index set and signatures.

[2] In [SBZ01], sanitizers are described as *owners*.

[3] "Sumi (Indian ink)" is a standard writing material in eastern Asian countries including Japan, and is used for non-digital sanitizations.

Table 1. A description of SUMI-4

Signer

1. For a given original document $\{m_i\}_{1\leq i\leq n}$, a signer padds random values r_i to each block.
2. For a padded document $M = \{(m_i, r_i)\}_{1\leq i\leq n}$, generate a set of hash values $H = \{h_i = \mathrm{Hash}(m_i, r_i)\}_{1\leq i\leq n}$ for a given hash function $\mathrm{Hash}(\cdot)$.
3. Generate a signature $s = \mathrm{Sign}_{\mathrm{signer}}(H)$ for a given signer's signing function $\mathrm{Sign}_{\mathrm{signer}}(\cdot)$.
4. Output (M, s) as an original document and a signature.

Sanitizer

1. Determine a disclosing index set $D \subset \{1, \ldots, n\}$, where blocks (m_i, r_i) $(i \in D)$ will be disclosed.
2. A sanitizer converts the document M to a new document $\tilde{M} = \{\tilde{m}_i\}_{1\leq i\leq n}$, where

$$\tilde{m}_i = \begin{cases} (m_i, r_i) & \text{if } i \in D \\ h_i & \text{if } i \notin D. \end{cases}$$

3. Output the sanitized document $\tilde{M}$ and the index set D.

Verifier

1. On input an original signature s, a sanitized document $\tilde{M} = \{\tilde{m}_i\}_{1\leq i\leq n}$, and a disclosing index set D, generate a set of hash values $H' = \{h'_i\}_{1\leq i\leq n}$, where

$$h'_i = \begin{cases} \mathrm{Hash}(\tilde{m}_i) & \text{if } i \in D \\ \tilde{m}_i & \text{if } i \notin D. \end{cases}$$

2. Compute $\mathrm{Verify}_{\mathrm{signer}}(H', s)$ and confirm the integrity of disclosed parts of the document for a signer's vefifying function $\mathrm{Verify}_{\mathrm{signer}}(\cdot)$.

Thus multiple sanitizations are possible in SUMI-2. But SUMI-2 is not efficient since it requires n signatures.

SUMI-3 generates n hash values (instead of signatures) corresponding to n-blocks and a signature on a concatination of these hash values. A sanitizer determines disclosing blocks and publishes a corresponding index set and the updated document which consists of original blocks for disclosing parts and hash values for closing parts. Thus multiple sanitizaions are possible in SUMI-3. Moreover, SUMI-3 is efficient. However, it is not secure becase the procedure is deterministic [MSI+03]. This idea is inherited to the following SUMI-4.

SUMI-4: In SUMI-4, in the beginning, all blocks are padded by random values to enhance the security. Then a set of hash values of all padded blocks and a signature on a concatination of these hash values are generated. When a sanitizer sanitizes a specified block, he/she replaces the block to the corresponding hash

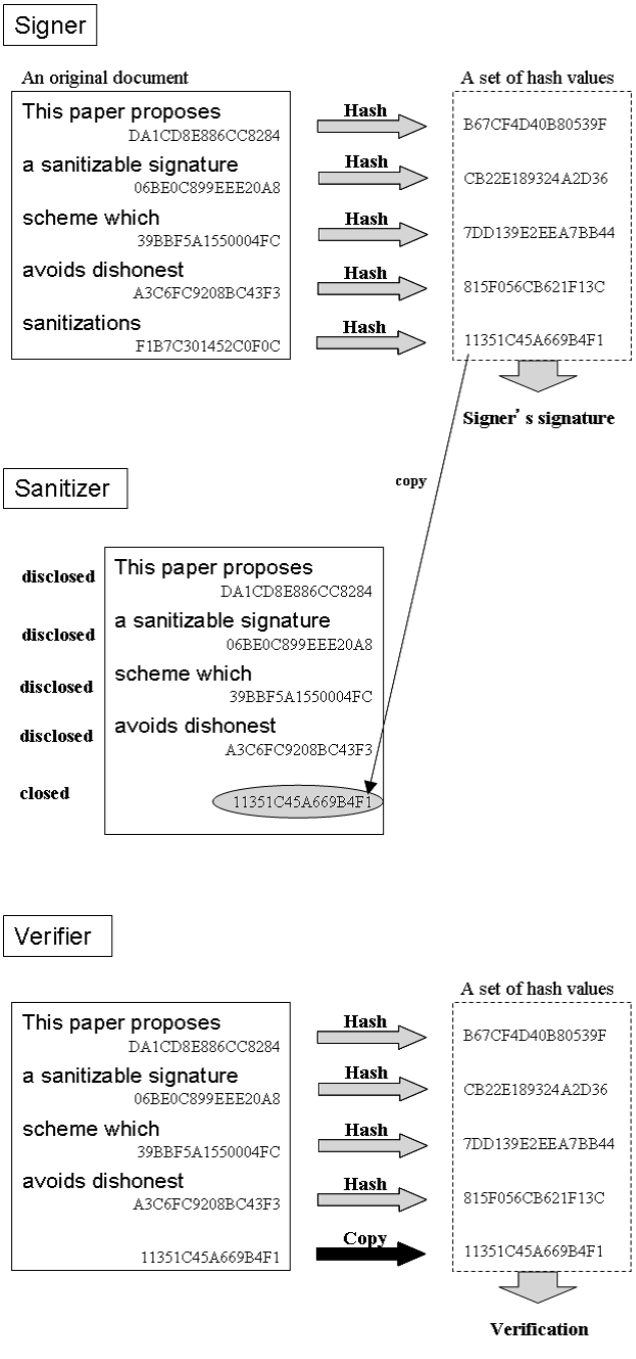

Fig. 1. An outline of SUMI-4

value. Thus multiple sanitizations are possible in SUMI-4. A formal description of SUMI-4 is in Table 1 (see also Figure 1).

Similar to SUMI-3, SUMI-4 is an efficient sanitizable signature scheme since it only requires 1 signature. Moreover, multiple sanitizations are possible in SUMI-4. However, SUMI-4 has the following security problem.

Additional Sanitization Attack: In all of SUMI-2, SUMI-3, SUMI-4, and CES schemes, multiple sanitizations are possible. In [MIM+05], Miyazaki et al. observed that this property allows an adversary to the *additional sanitization attack*, a variant of the man-in-the-middle attack, in which the adversary intercepts an appropriately sanitized document, and sends the corruptly sanitized document to a verifier to which dishonestly sanitizations are added. This is because sanitizers are anonumous in these schemes; in order to resist this kind of attacks, anonymous sanitizations should be avoided.

SUMI-5: Miyazaki et al. proposed an enhanced sanitizable signature scheme SUMI-5 in the same paper [MIM+05]. Since SUMI-5 is based on SUMI-4, SUMI-5 is also very efficient and multiple sanitizations are possible. Moreover, adversaries' dishonestly sanitizations are avoided. A formal description of SUMI-5 is in Table 2.

In SUMI-5, dishonest sanitizations are avoided by using three sets D_{n}, D_{na}, C. By changing these sets, multiple sanitizations are possible, however, since each sets should be monotonously increasing or decreasing, latter sanitizers can prohibit sanitizations beyond the disclosing conditions determined by previous sanitizers. On the other hand, because the signer cannot determine any conditions on the sanitization, the first sanitizer cannot determine the disclosing condition.

SUMI-5 is a very interesting scheme because it firstly excludes dishonest sanitizations. However, since a closing condition should be determined in the beginning, it seems impractical. Our study is motivated by this problem. Similar to SUMI-5, our proposed scheme "PIATS" is based on SUMI-4 rather than SUMI-5, but proceeds another direction as in the next section.

3 Proposed Scheme

In this section, we propose a new partially sanitizable signature scheme "PIATS" (Partial Information Assuring Technology for Signature) which supports multiple sanitizations with avoiding dishonest sanitizations. In order to resist the additional sanitization attack and solve the problem of SUMI-5 (discussed in the previous section), we establish three conditions which the sanitizable signature scheme should satisfy:

(C1) Signers cannot determine disclosed blocks.
(C2) Sanitizers and their sanitizated parts can be identified by verifiers.
(C3) Sanitizers cannot control other blocks than their sanitizing blocks.

Note that SUMI-5 satisfies the condition (C1) and (C2), but not (C3). This may be a main reason why SUMI-5 requires those severe limitations on index sets.

Table 2. A description of SUMI-5

Signer

1. For a given original documnet $\{m_i\}_{1\leq i\leq n}$, a signer padds random values r_i to each block. Also generate random values s_i for each block. Let $M = \{(m_i, r_i)\}_{1\leq i\leq n}$ be a padded document and $S = \{s_i\}_{1\leq i\leq n}$ be a set of random values.
2. For an i-th block, draw a line ℓ_i passing two points $(1, \mathrm{Hash}(m_i, r_i))$ and $(2, \mathrm{Hash}(s_i))$ for a given hash function $\mathrm{Hash}(\cdot)$. Next, compute two values $P_i,\ Q_i$ such that two points $(0, Q_i),\ (3, P_i)$ are on the line ℓ_i. Let $P = \{P_i\}_{1\leq i\leq n}$ be a set of P_i values.
3. Generate a signature $s = \mathrm{Sign}_{\mathrm{signer}}(Q_1||\ldots||Q_n||P_1||\ldots||P_n)$ for a signer's signing function $\mathrm{Sign}_{\mathrm{signer}}(\cdot)$, where $||$ denotes a concatination.
4. Output (M, S, P, s) as an original document and a signature.

Sanitizer

1. Determine three index sets (partitions) $D_{\mathrm{a}},\ D_{\mathrm{na}},\ C \subset \{1, \ldots, n\}$ such that $D_{\mathrm{a}} \cup D_{\mathrm{na}} \cup C = \{1, \ldots, n\}$ and $D_{\mathrm{a}} \cap D_{\mathrm{na}} = D_{\mathrm{na}} \cap C = C \cap D_{\mathrm{a}} = \phi$. For an index $i \in D_{\mathrm{a}}$, the i-th block is disclosed and additional sanitization is allowed. For an index $i \in D_{\mathrm{na}}$, the i-th block is disclosed but additional sanitization is not allowed. On the other hand, for an index $i \in C$, the i-th block is closed at all times.
2. Let $\tilde{M} = M\backslash\{(m_i, r_i)\}_{i\in C}$, $\tilde{S} = S\backslash\{s_i\}_{i\in D_{\mathrm{na}}}$.
3. Output the sanitized document $(\tilde{M}, \tilde{S})$ with the index sets $D_{\mathrm{n}},\ D_{\mathrm{na}},\ C$.

Verifier

1. On input an original signature s, a sanitized document $\tilde{M} = \{\tilde{m}_i\}$ with sets $\tilde{S},\ P$, and an index set C, for each block, draw a line ℓ'_i passing two points $(1, \mathrm{Hash}(\tilde{m}_i))$ and $(3, P_i)$ if this block is disclosed or $(2, \mathrm{Hash}(s_i))$ and $(3, P_i)$ if this block is closed.
2. For each line ℓ'_i, compute Q'_i such that a point $(0, Q'_i)$ is on the line ℓ'_i.
3. Compute $\mathrm{Verify}_{\mathrm{signer}}(Q'_1||\ldots||Q'_n||P_1||\ldots||P_n, s)$ and confirm the integrity of disclosed parts of the document for a signer's vefifying function $\mathrm{Verify}_{\mathrm{signer}}(\cdot)$.

3.1 Approach

After analyzing the previous sanitizable signature schemes, we reached a conclusion that a main reason why dishonest sanitizations are possible is because the anonymity of the sanitizers, namely there are no identifications in sanitizations. We consider that identifications of a signer and sanitizers are the most required property for secure sanitizable signature schemes.

In order to establish a new scheme, we went back to SUMI-4 rather than SUMI-5. In SUMI-4, a signature scheme assures the integrity of a hash set H, and some properties of the hash function (the preimage resistance and the 2nd

preimage resistance) assures the integrity of the sanitized document. In other words, previous sanitizable signature schemes generate a sign on a hash value set H and verify the integrity of H, while standard (non-sanitizable) signature schemes generate a sign on a document M and verify the integrity of M. With this property, sanitizable schemes can verify the integrity of the sanitized document by veifying the integrity of H without a direct accesse on the document.

3.2 Description of the Proposed Scheme

This section describes a proposed sanitizable signature scheme "PIATS", in which a signature on an original document is generated from a hash value set H and its signature. Closing blocks and corresponding padded random values are replaced by distinct characters (such as "XXXXXXXX" for example) and by new random values, respectively, in sanitizations. Corresponding hash values are also recomputed. Thus a new hash value set H' and its signature s' is obtained. Then the sanitizer publishes the sanitized document and the sanitizer's signature s' in addition to the original signature s. On input these data, a verifier verifies the integrity of the sanitized document. In addition, sanitized blocks can be identified by comparing two sets H and H'. Finally, from a disclosed document and the sanitized index sets, the verifier can verify the correctness of the sanitizings. An formal description of the proposed sanitizable signature scheme is in Table 3 (see also Figure 2)D We name the scheme as the Partial Information Assuring Technology for Signature (PIATS).

By the described procedures, a verifier can verify the integrity of disclosed blocks, identify the sanitized parts and sanitizers. If a closed block (combined with former and latter blocks) has a meaning, it can be treated as either a valid sanitization or a forgery depending on the policy. Thus PIATS allows multiple sanitizations with avoiding dishonest sanitizations.

3.3 Security Analysis

Let us consider the security of the proposed scheme. The unforgeability (no forgeability of signatures), secrecey (no leakage of sanitized information), and the integrity (no forgery on unsanitized parts with valid verification), is established as in the following observations.

Unforgeability: Combined with secure digital signature schemes such as RSA-PSS [PKCS], any forgeries can be avoided in the proposed scheme.

Secrecy: Since a signer's signature S consists of a set of hash values and its signature, and a sanitizer's signature S' consisnts of an updated set of hash values and its signature, the secrecy of the sanitized information is assured by the preimage resistance of the hash function.

Integrity: In the proposed scheme, by comparing a hash value set generated by a signer and by a sanitizer, dishonest sanitizations can be identified by verifiers. This is assured by the 2nd preimage resistance of the hash function.

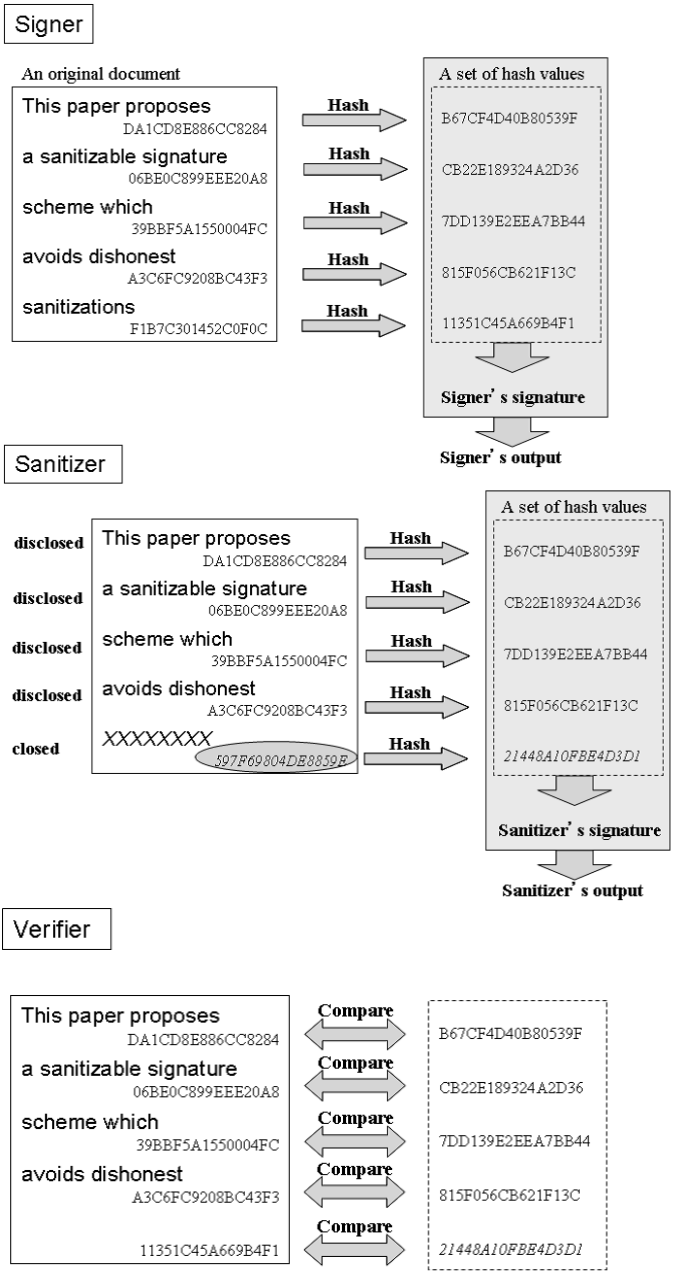

Fig. 2. An outline of the proposed sanitizable signature scheme "PIATS"

Table 3. A description of the proposed sanitizable signature scheme "PIATS"

Signer

1. For a given original documnet $\{m_i\}_{1\leq i\leq n}$, a signer padds random values r_i to each block.
2. For a padded document $M = \{(m_i, r_i)\}_{1\leq i\leq n}$, generate a set of hash values $H = \{h_i = \mathrm{Hash}(m_i, r_i)\}_{1\leq i\leq n}$ for a given hash function $\mathrm{Hash}(\cdot)$.
3. Generate a signature $s = \mathrm{Sign}_{\mathrm{signer}}(H)$ for a signer's signing function $\mathrm{Sign}_{\mathrm{signer}}(\cdot)$.
4. Set $S = H||s$ where $||$ denotes a concatination.
5. Output (M, S) as an original document and a signature.

Sanitizer

1. Determine a disclosing index set $D \subset \{1, \ldots, n\}$, where blocks (m_i, r_i) $(i \in D)$ will be disclosed.
2. A sanitizer converts the document M to a new document $\tilde{M} = \{\tilde{m}_i\}_{1\leq i\leq n}$, defined by

$$\tilde{m}_i = \begin{cases} (m_i, r_i) \text{ if } i \in D \\ (m'_i, r'_i) \text{ if } i \notin D, \end{cases}$$

where m'_i is a distinct characters (such as "XXXXXXXX") and r'_i is a random value for padding.
3. Generate a set of hash values $H' = \{h'_i = \mathrm{Hash}(\tilde{m}_i)\}_{1\leq i\leq n}$. Then generate a new signature $s' = \mathrm{Sign}_{\mathrm{sanitizer}}(H')$ for a signing function $\mathrm{Sign}_{\mathrm{sanitizer}}(\cdot)$ and set $S' = H'||s'$
4. Output the sanitized document and a signature $(\tilde{M}, S')$ with the index set D.

Verifier

1. On input an original signature s, the sanitized document and a signature $(\tilde{M}, S')$, recover H, s, H', s'.
2. Compute $\mathrm{Verify}_{\mathrm{signer}}(H, s)$ and confirm the integrity of the original document for a vefifying function $\mathrm{Verify}_{\mathrm{signer}}(\cdot)$.
3. Compute $\mathrm{Verify}_{\mathrm{sanitizer}}(H', s')$ and confirm the integrity of the sanitized document and identify the sanitizer for a vefifying function $\mathrm{Verify}_{\mathrm{sanitizer}}(\cdot)$.

3.4 Multiple Sanitization

The proposed scheme allows multiple sanitizaions by adding sanitizers' signatures. Let $(M^{(0)}, S^{(0)})$ be a pair of the original document and a signer's signature on $M^{(0)}$. Similarly, let $(M^{(j)}, S^{(j)})$ be a pair of the j-th sanitized document and the j-th sanitizer's signature on $M^{(j)}$ for $1 \leq j \leq k$. Here k is the admissible number of sanitizers determined in advance as a security parameter of the scheme. Then the last (k-th) sanitizer publishes $(M^{(k)}, S^{(0)}, S^{(1)}, \cdots, S^{(k)})$.

From the published information, a verifier can identify which blocks are sanitized by the j-th sanitizer ($1 \leq j \leq k$) and which blocks are signed by the signer.

The proposed scheme has a property that the number of disclosing blocks can be increasing. This property my be required in most situations where sanitizable signature schemes are used. Conversely and interestingly, the proposed scheme also has a property that the number of disclosing blocks can be decreasing, if all sanitizers can access on the original documet.

4 Comparison

In this section, we compare sanitizable signature schemes including CES-CV [SBZ01], SUMI-4 [MSI+03], SUMI-5 [MIM+05], and the proposed scheme PIATS from viewpoints of the ability of multiple sanitizations, required conditions for future sanitizations, and combinable signature schemes. A comparison is summarized in Table 4. Note that as described in section 2, CES-CV and CES-HT are potentialy identical to SUMI-4.

CES-RSAP and CES-MERP are combined to only specified (RSA-type) signature schemes, while other schemes can be combined with arbitrary secure signature schemes. All sanitizable schemes but SUMI-1 support multiple sanitizations. A main defference between SUMI-2, SUMI-3, SUMI-4 is the number of required signatures; for signing an n-block document, SUMI-2, SUMI-3, and SUMI-4 requires $2^n,\ n,\ 1$ signatures, respectively. Thus SUMI-4, its successor SUMI-5, and PIATS are efficient with regard to the number of required signatures. Only SUMI-5 and PIATS can avoid dishonest sanitizations. However, as described in section 2, SUMI-5 requires a rather impractical assumption in which a closing policy should be determined in the beginning. On the other hand. PIATS avoids dishonest sanitizations without such a limitation.

Table 4. A comparison of sanitizable signature schemes

Scheme	Multiple Sanitization		Combinable Signature Scheme	Specifying Future Sanitizations
	(Honest)	(Dishonest)		
CES-CV CES-HT	Possible	Possible	Arbitrary	Unrequired
CES-RSAP CES-MEPR	Possible	Possible	Limited (RSA-type) (RSA-type)	Unrequired
SUMI-1	Impossible	Impossible	Arbitrary	—
SUMI-2	Possible	Possible	Arbitrary	Unrequired
SUMI-3	Possible	Possible	Arbitrary	Unrequired
SUMI-4	Possible	Possible	Arbitrary	Unrequired
SUMI-5	Possible	Partially possible	Arbitrary	Required
PIATS	**Possible**	**Impossible**	**Arbitrary**	**Unrequired**

5 Concluding Remarks

This paper proposes a new partially sanitizable signature scheme PIATS which support multiple sanitizations with avoiding dishonest sanitizations, and therefore enables to manage documents with secure and privacy-protected. Obviously, the proposed scheme is suitable for managing digital documents (described in XML format for example). However, applying PIATS to scanned documents is not easy, because scanned documents are recorded in picture formats. Since there are so many types of formats, we have to consider how to apply PIATS to each format separately. Recently, we developed an experimental system which manages a sanitizable signatue schemes on jpeg formats. Intuitively, the system works well, but there are many problems to overcome. Further experiments and analysis will be required to use PIATS in practical systems.

Acknowledgments

The authors would like to thank Naoya Torii, Satoru Torii and Takeshi Shimoyama for their helpful comments and suggentions on the early version of this paper.

References

[ACM+05] G. Ateniese, D.H. Chou, B. Medeiros, G. Tsudik, "Sanitizable Signatures", *ESORICS 2005*, LNCS 3679, pp. 159–177, Springer-Verlag, 2005.

[MIM+05] K. Miyazaki, M. Iwamura, T. Matsumoto, R. Sasaki, H. Yoshiura, S. Tezuka, H. Imai, "Digitally Signed Document Sanitizing Scheme with Disclosure Condition Control", The Institute of Electronics, Information and Communication Engineers (IEICE) Trans. on Fundamentals, Vol, E88-A, pp. 239–246, No. 1, January 2005. Available from `http://search.ieice.org/index-e.html`

[MSI+03] K. Miyazaki, S. Susaki, M. Iwamura, T. Matsumoto, R. Sasaki, H. Yoshiura, "Digital Documents Sanitizing Problem", The Institute of Electronics, Information and Communication Engineers (IEICE) technical report, ISEC 2003-20, May 2003.

[PKCS] RSA Laboratories, "PKCS #1 v2.1: RSA Encryption standard",June 14, 2002,Available from `http://www.rsasecurity.com/rsalabs/node.asp?id=2125`

[SBZ01] R. Steinfeld, L. Bull, and Y. Zheng, "Content Extraction Signatures", *ICICS 2001*, LNCS 2288, pp. 285–304, Springer-Verlag, 2001.

[Wir00] "Carnivoe Review Team Exposed!",an article of Wired News Reports, 2000. Available from `http://www.wired.com/news/politics/0,1283,39102,00.html`

[Wir02] "NYT Site Exposes CIA Agents", an article of Wired News Reports, 2002. Available from `http://www.wired.com/news/politics/0,1283,37205,00.html`