Content Control Scheme to Realize Right Succession and Edit Control

Katsuma Koga[1], Masaki Inamura[2], Kitahiro Kaneda[3] and Keiichi Iwamura[1]
[1]*Tokyo University of Science, Niijuku 6-3-1, Katsusika-ku, Tokyo, Japan*
[2]*Tokyo Denki University, Ishizaka, Hiki-gun, Hatoyama-machi, Saitama, Japan*
[3]*Osaka Prefecture University, Sakai-shi, Osaka, Japan*

Keywords: Copyright Protection, Right Succession, Edit Control, Electronic Signature, Aggregate Signature.

Abstract: We propose a copyright protection technology suitable for consumer generated media such as You Tube and CLIP. This technology realizes right succession of and edit control by the previous work's authors. In this technology, we use a digital signature to confirm the relation between the primary and secondary authors and to determine whether the contents may be edited. We propose to apply this technology to CLIP, which is a software used to operate a pre-set character three-dimensional (3D) model of a three-dimensional computer graphics (3DCG) for creating computer animation. In addition, we provide three security methods for the proposed technologies.

1 INTRODUCTION

Because of the widespread use of the Internet, circulation of content created by consumers has increased. This consumer content is known as consumer generated media (CGM) and all Internet users can be content creators and distributors. You Tube and CLIP are well-known and widely used examples of CGM services.

However, in conventional content distribution by high-quality TV and DVD, etc., viewing and copy control are used as copyright-protection technology. However, viewing control is unsuitable for CGM services, because most authors who produce CGM content want to have their content widely viewed and without restrictions. Copy control is also useless because most authors want their content to be used in other content forms.

Therefore, new copyright-protection technologies are required by CGM services. We propose that right succession and edit control are suitable as new copyright protection technologies for CGM services.

Right succession is a technology that protects an author's copyright when his or her work is used in a secondary source (Tetsuya, 2007). This is realized by introducing a signature, which shows the relation between the primary- and secondary-content authors. A verifier can verify the signature and relation between the primary and secondary authors.

Edit control is a technology of permitting editing primary content in secondary contents that permits a partial content edited as secondary content to the contents by the author. An author creates and reveals a signature that gives permission for edits to any partial primary content. An editor can then edit that content. The verifier confirms that the editor is editing only that content to which the author gave permission. This technology has the advantage of reducing time and effort for an editor in setting permissions, because editing permissions established by an author can be saved by an editor. The editable content is shown in advance by the author.

These two independent technologies of right succession and edit control have been proposed in other studies (Tatsuhiko, 2011) and (Masaki, 2012). However, because the two technologies use different signatures, they may be incompatible. Therefore, in this study, we propose a technology that realizes the two simultaneously. Usage is I will be described later. This technology can also be applied to cases in which two or more authors collaborate on a single work. In addition, we provide three security methods concerning prevention of aggregate signature forging, disguised participation, and collusion attacks.

The remainder of this paper is structured as follows. We describe applied services in Section 2 and related research in Section 3. We describe our

Koga K., Inamura M., Kaneda K. and Iwamura K..
Content Control Scheme to Realize Right Succession and Edit Control.
DOI: 10.5220/0005535602490256
In *Proceedings of the 12th International Conference on e-Business* (ICE-B-2015), pages 249-256
ISBN: 978-989-758-113-7

proposed scheme for realizing the two technologies in Section 4 and the three security methods in Section 5. The last section provides a conclusion.

2 SERVICES PROPOSED

Section 1 referred to specific CGM services such as You Tube and CLIP. CLIP is a web site to support creating animation by consumers, and to operate a pre-set character 3D model of 3DCG. In CLIP, consumers can reuse some exhibited works in their own work. Our proposed scheme is suitable for the service like CLIP which makes the work using the prepared materials, because our scheme can specify the right relation between authors and the portion which permits edit in advance, when a work is reused.

When using this scheme, the content and the digital signature need to be connected using an digital watermark. If the digital watermark is robust against an attack to the history information on signatures, the content creator rights including editor are guaranteed. Although CLIP is managed by an administrator, our scheme realizes continuous copyright management in two or more sites. In addition, if the player machine mentioned later is standardized, our scheme realizes a new copyrights protection which does not need an administrator.

3 RELATED RESEARCH

3.1 Edit Control

Sano et al. proposed an digital signature scheme that can control the edits of content in advance. In this paper, edit includes modify and delete a part of contents (Kunihiko, 2008). In their proposed scheme, the author divides his content into each area to be edited, and gives the signature for controlling edit to each area.

When permitting edit, the signature is opened to the public. When not permitting edit, the signature serves as nondisclosure. When editor edit the permitted areas, he substitutes the signature for a new signature for the edited contents. Editor cannot edit it, when the signature is not exhibited, because he cannot exchange the signature for a new signature. This scheme not only allows the consumer to edit permitted area of contents, it can also add new contents in the perming area by adding new signature.

As mentioned above, the scheme by Sano et al. can carry out prior control of the edit of contents. However, since this scheme treated signatures of an author and an editor equally, ordering of the author and the editor had not been completed. Therefore, a problem arises in which the relationship between the author and editor is unclear. In addition, this technique assumes the case where there is an author. Thus, it does not consider cases in which multiple authors produce content.

3.2 Right Succession

Inamura et al., proposed an expression of a quotation process in contents with a new tree-structure-specified aggregate signature scheme. This proposed scheme solves right succession. This uses a new tree-structure representation-type aggregate signature based on a GDH group. The tree-structure-specified aggregate signature scheme can describe not only who signs but also which division each of signers belongs to. The new tree-structure represents a process until finally can content. The method of right succession is multiplied by the electronic information (Katsuhiko, 2006). It is that secondary producer is the signature for the previous content. In other words, keys of the previous human and the content of the later human is guaranteed rights information multiplied. Accordingly, this signature indicates that the ordered aggregate signature (Boneh, 2003). can be realized by being synthesized many times. Because this approach extends the steps in one-to-many, stacking the relationship can be extended to the aggregate signature of the tree-representation type (Masaki, 2009). However, this approach does not consider the configuration of the editing of content. Therefore, constitute the aggregate signature for each of the content editing, it is necessary to fix the beginning.

4 THE PROPOSED SCHEME

The proposed scheme realizes right succession and edit control simultaneously (Yamamoto, 2006). Multiple of the author is potluck their own content, I can perform a pre-edit control. Editors to propose a scheme that can create a secondary content based on it.

In this scheme, we assume two entities having the following roles.

Author: creates content and sets to edit control.

If multiple authors want to produce a single content, the hierarchy is that of primary author

(Masaki, 2010). followed by the secondary author, and then the division of editors. The i following author by i + 1 primary author is an author for creating content as a content editing of the original. The primary author creates original content which is first content.

Verifier: verifies whether the content contains a legitimate signature.

Each author has his own secret key as well as a public key, must sign the content they created, and edit any data as his preference that has been divided. In addition, they must retain the copyright (to change, delete, add content), which may be one of the controls set in advance. The editor has a public key and his or her own secret key pertaining to editing privileges (for editing data in the parts that are allowed). Verifying public keys of editors and authors require signature verification. The verifier provides a content playback device that may include content without a valid signature and was thus produced from a scheme that the content cannot be played.

4.1 State Transition

State transition is to edit control at divided data. This section describes a state transition for setting the conditions to realize editing control of the author's content data. However, the data are considered to be of two types: existing and empty. Existing data which has a real data represents a portion of the content configuration data and have the following conditions (a)–(f). In contrast, empty data has no real data. Empty data are considered control data for controlling the additional aspects (g) and (h) of state transition which is making the edit control on the content. Empty data which doesn`t has a real data is changed to existing data, which the i following author (i hierarchy on the author) produces. The state transitions are shown below.

(a) can be changed, can be deleted

(b) can be changed, cannot be deleted

(c) cannot be changed, can be deleted

(d) cannot be changed, cannot be deleted

(e) can be changed or deleted

(f) cannot be changed or deleted

(g) can be added

(h) cannot be added

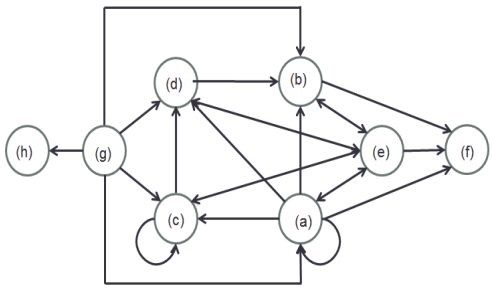

Figure 1: The state transition.

4.2 Control State

A pre-control example of each Data is shown in Fig. 2.

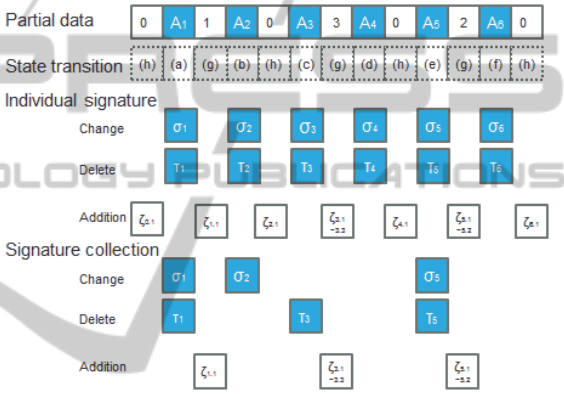

Figure 2: Pre-control example.

Control state in three partial data, two types with existing data, using one type of signature empty data. In addition, a control state can change, delete, and add data to realize the control state. Authors of existing data are modified for individual change signing σ_i and individual delete signing τ_i by the private key of the author. The secondary authorship constitutes an ordered aggregate signature (Boneh, 2003). To indicate the order of the primary and secondary authors, we use a hash value for the deletion and for changing the intermediate signature created. In addition, this operation is a form of tree-structure representation because it performs in a multistage manner. The empty data creates an additional signature ξ_i using the private key of the author. In addition, each set of data has a unique signature to prove that actual creator created the content. Moreover, authors are allowed to edit or publish the signature as public information in the signature collection. Each changed, deleted, and added signature collection to prepare G_c, G_d, G_a corresponding to the additional. These are input as individual signatures at portions in which state

transition is allowed. In contrast, if it does not allow editing, not signed output of the content in each set.

4.3 Algorithm

Fig. 2 shows an example algorithm for the control state of the scheme. This scheme has the following prerequisites to meet the security model (Komano, 2005) and (Komano, 2008).

Algorithm:

1. A public key infrastructure (PKI) (Helena, 2007). has been developed. In addition, all of the signers of the signing and verification key pair have been duly issued by a certificate authority (CA).
2. In addition to having a key pair that is issued to the signer, the scheme does not issue a new key pair.
3. In aggregate during signature generation, communication between the signers is impossible for the purpose of obtaining third-party intermediate information being created conducted safely.
4. The relationship of the content and digital signature is integral to sign the content so that a secure electronic watermark cannot be separated.

As Fig. 2 shows, content is divided into six parts, from 1 to 3 in the order of author creation the primary content creates content when they edit and synthesize content. Multiple authors showing an example in which going edited and synthesized, generally has role or sharing is defined in each author. The i-order author, by combining the partial contents i-1 order author has made, will complete the content. Using the example of animation, multiple primary authors can collaborate to produce a coma, which is a small part of anime. The secondary author can synthesize it. Work such as music can also be performed and the tertiary author can synthesize pictures and music. Therefore, the scheme is set up so that the author control state is already meeting.

4.4 Algorithm

Key Generation:

A generator is defined as $g \in G_1$. A signer selects $s_{ij} \in Z_p^*$ (all of the signer's signature keys (private key) will be with each different from them), and calculates $V_{ij} = s_{ij}g$.

Signature Generation:

(1) Original content A can be divided into areas $A_1 \sim A_6$, for edit control.

$$A \to A_1 \sim A_6 \tag{1}$$

(2) Content identifier ID, which generates the part of content identifier ID_i, is set as follows (i = 1, ⋯, 6).

$$A_i * = ID||ID_i||A_i \tag{2}$$

(3) One signature is set in front of the existing data A_1, and the others are set between $A_i$and A_{i+1} when the addition of empty data is permitted. Number of empty data is $B_{i-0.5}$, in which the default is set as follows by using the dummy data d.

$$B_{i.-0.5} * = ID||ID_{i-0.5}||d \tag{3}$$

(4) Each set of existing and real data in the state transition determines the state (a)–(d). In addition, empty data from the state transition determines the state (g) and (h) .

(5) From $A_i *$ and $B_{i-0.5j} *$, we generate a hash value.

$$h_i = \boldsymbol{H(}A_i\boldsymbol{*)},\ h_{i.j} = \boldsymbol{H(}B_{i-0.5j}\boldsymbol{*)} \tag{4}$$

(6) Individual signature are generated for change, deletion, and addition of control

$$\text{Chang:}\sigma_i = H(ID||ID_i||h_i||0^c)^{s_i} \tag{5}$$

$$\text{Deletion: } \tau_i = H(ID||ID_i||h_i||1^c)^{s_i} \tag{6}$$

$$\text{Addition: } \zeta_{i-0.5} = H(ID||ID_{i-0.5}||d||0^c)^{s_i} \tag{7}$$

Author generates the following to produce the ordered signatures.

$$\delta_{1j} = s_{1j}h_{1j} \tag{8}$$

In the content, to control change, deletion, and addition, $\sigma_i, \tau_i, \zeta_{ij}$ output the allowed partial data $\mathrm{G}_c, \mathrm{G}_d, \mathrm{G}_a$ content.

Secondary author: second-order authors confirm partial data $\mathrm{G}_c, \mathrm{G}_d, \mathrm{G}_a$ and the editorial content in accordance with the primary author's editing control set, whereby

$$\delta_{21} = s_{21}h_{21} + \sum_{j=1}^{4}(\sigma_{1j} + \tau_{1j}) + s_{21}\sum_{j=1}^{4}(x_{1j} + y_{1j}) \tag{9}$$

$$\delta_{22} = s_{22}h_{22} + \sum_{j=5}^{2}(\sigma_{1j} + \tau_{1j}) + s_{22}\sum_{j=5}^{2}(x_{1j} + y_{1j}) \tag{10}$$

and thus create each aggregate signature δ_{21}, δ_{22}. In the content, to control change, deletion, and addition, $\sigma_{2j}, \tau_{2j}, \zeta_{2j}$ output the allowed partial data G_c, G_d, G_a content. In order to updates the previous signature, the ordered aggregate signature subtracts it, adding its own updated signature.

The tertiary author (the last editor): third-order authors confirm partial data G_c, G_d, G_a and the editorial content in accordance with the primary or second author`s editing control set,

$$\delta_{31} = s_{31}h_{31} + \sum_{j=1}^{2} \delta_{2j} + s_{31} \sum_{j=1}^{2} h_{2j} \qquad (11)$$

which creates an aggregate signature δ_{31}.

Content A*={A_1*,・・・, A_6*},empty data $B_{i.-0.5}$*, edit permit data individually signing set all components of G_c, G_d, G_a, and these output an ordered aggregate signature. All of this is public information.

Data Update:

When the editor changes the partial data A_i* to N_i* (A_i*→N_i*), the following process (i)–(v) is defined. In addition, the expression for changing partial data (9) and (10) is defined as shown in the procedures of (12)–(15). In addition, the edited order aggregate are $\delta`_{21}, \delta`_{31}$. When changing A_i* to N_i*, the hash of changed content is defined $h`_{21}, h`_{31}$.

After setting (a) is changed:

$$\left.\begin{array}{c} \delta`_{21} = \delta_{21} - s_{21}h_{21} + s_{21}h`_{21} \\ -(\sigma_{11} + \tau_{11}) + (\sigma`_{11} + \tau`_{11}) \\ \delta`_{31} = s_{31}h`_{31} + \delta`_{21} + \delta_{22} \\ +s_{31}(h`_{21} + h_{22}) \end{array}\right\} \qquad (12)$$

After setting (b) is changed:

$$\left.\begin{array}{c} \delta`_{21} = \delta_{21} - s_{21}h_{21} + s_{21}h`_{21} \\ +\sigma_{11} - \sigma`_{11} \\ \delta`_{31} = s_{31}h`_{31} + \delta`_{21} + \delta_{22} \\ +s_{31}(h`_{21} + h_{22}) \end{array}\right\} \qquad (13)$$

After setting (c) is changed:

$$\left.\begin{array}{c} \delta`_{21} = \delta_{21} - s_{21}h_{21} + s_{21}h`_{21} \\ +\tau_{11} - \tau`_{11} \\ \delta`_{31} = s_{31}h`_{31} + \delta`_{21} + \delta_{22} \\ +s_{31}(h`_{21} + h_{22}) \end{array}\right\} \qquad (14)$$

After setting (d) is changed:

$$\left.\begin{array}{c} \delta`_{21} = \delta_{21} - s_{21}h_{21} + s_{21}h`_{21} \\ -(\sigma_{11} + \tau_{11}) + (\sigma`_{1.11} + \tau`_{1.11}) \\ \delta`_{31} = s_{31}h`_{31} + \delta`_{21} + \delta_{22} \\ +s_{31}(h`_{21} + h_{22}) \end{array}\right\} \qquad (15)$$

(i) Change: A_{ij} *→ N_{ij} *, $h'_i = H(N_{ij}*)$, $\sigma'_{ij} = H(ID||ID_{ij}||h'_{ij}||0^c)^{s_{ij}}$

(ii) Deletion:$\tau'_{ij} = H(ID||ID_{ij}||h'_{ij}||1^c)^{s_{ij}}$,

(iii) Hash value difference: $\varepsilon_{ij} = H(ID||ID_i||h'_{ij} - h_{ij}||1^c)^{s_{ij}}$

(iv) Addition of change case: $B_{i-0.5,j}$*←$N_{i-0.5,j}$*, $h'_{i-0.5,j}$'=H($N_{i-0.5,j}$*), $\xi_{i-0.5,j}/\{\xi_{i-0.5,j}\}$,

$$\sigma'_{i,j} = H(ID\,||ID_{i-0.5,j}||\,h'_{i-0.5,j}||1^c)^{s_ij}$$

(v) Addition of deletion case:

$$\tau'_{i,j} = H(ID\,||ID_{i.j}||\,h'_{i.j}||1^c)^{s_i}$$

1. Determine edited partial data of A_i* or $B_{i.-0.5}$ * from original content A*.
2. Depending on the type of editing conducted, the following processing is performed. However, A_{ij}* represents the original partial data, N_{ij}* is change data,$B_{i-0.5.j}$ * is additional empty data, $N_{i-0.5.j}$ * is exiting data from the previous empty data and (j = 0, ..., 6).

Control Change Pattern:

(a)→(a): (i), G_c←$G_c \cup \sigma'_{ij}$, (ii), (12), G_d←$G_d \cup \tau'_i$

(a)→(b): (i), (12), G_c←$G_c \cup \sigma'_{ij}$, (ii)

(a)→(c): (i), (ii), (12), G_d←$G_d \cup \tau'_{ij}$

(a)→(d): (i), (ii), (12)

(a)→(e): N_{ij}* as empty data (i), G_c←$G_c \cup \sigma'_{ij}$', (ii), (12), G_d←$G_d \cup \tau'_{ij}$ (If the deletion of data of the position is disallowed, do not perform G_d←$G_d \cup \tau'_{ij}$)

(a)→(f): N_{ij}* as empty data (i), (ii), (12)

(b)→(b): (i), G_c←$G_c \cup \sigma'_{ij}$, (iii), G_{bd}←$G_{bd} \cup (h'_{ij} - h_{ij})$, (13)

(b)→(d): (i), (iii), G_{bd}←$G_{bd} \cup (h'_{ij} - h_{ij})$, (13)

(c)→(d): $G_d \setminus \{\tau_{ij}\}$, (14)

(c)→(f): N_{ij}* as empty data A_{ij}*→N_{ij}*, h'_{ij} = H(N_i*), $\sigma/\sigma_{ij}, G_c \setminus \{\sigma_{ij}\}$, (iii), G_{bc}←$G_{bc} \cup (h'_{ij} - h_{ij})$, (ii), (14)

(e)→(a): (i), G_c←$G_c \cup \sigma'_{ij}$, (ii), G_d←$G_d \cup \tau'_{ij}$, (12)

(e)→(b): (i), G_c←$G_c \cup \sigma'_{ij}$, (ii), (12)

(e)→(c): (i), (ii), G_d←$G_d \cup \tau'_{ij}$, (12)

(e)→(d): (i), (ii), (12)

(g)→(a): (iv), G_c←$G_c \cup \sigma'_{i,j}$,(v), G_d←$G_d \cup \tau'_{i,j}$, (15)

(g)→(b): (iv), G_c←$G_c \cup \sigma'_{i,j}$', (v), (15)

(g)→(c): (iv), (v), G_d←$G_d \cup \tau'_{i,j}$, (15)

(g)→(d): (iv), (v), (15)

(g)→(h): (15)

3. ε_i is aggregated, empty set data ε, to generate the signature σ_r and editing historical data r.

$$\varepsilon = \varepsilon_i \tag{16}$$

4. Updated content, individual signature collection G_c, G_d, G_a aggregate signature $\delta`_{31}$ or δ_{31} and ε, hash difference value G_{bd}, G_{bd} and editing history data r. Its σ_r its signature, all are output to the verifier.

Verification:

I. Verify articulated content identifier ID in the partial data is equal to A *.

II. In addition, the verifier generates the following from additional empty data $B_{i.j}$* that has not been added. In addition, $\zeta \leftarrow \sum_i^6 \zeta_{ij}$ is verified from additional data ζ_{ij}.

III. If verification is correct, to verify any omissions to the ID of the partial data. Partial data identifier ID_i of all partial data A_i is used to verify whether the data are in ascending order, with the exception of the order of additional empty data if verification is correct.

IV. We confirm that the actual data exists with a hash value difference in G_{bd} and that the actual data does not exist with the hash value difference in G_{bc}. If they are correct, we verify the signature using the hash value difference.

$$\left.\begin{aligned} w_i &= H(ID \left|\left| ID_{ij} \right|\right| h'_{ij} - h_{ij} || c^1) \\ w &= \prod w_{ij}, e(\sum \varepsilon_{ij}, g) = e(w, \sum V_{ij}) \end{aligned}\right\} \tag{17}$$

V. If verification is correct and it generates a hash value h'_{ij} =$H(A_{ij}$ *) of the partial data, it calculates a hash value for the original data.

$$h'_{ij} - \left(h'_{ij} - h_{ij}\right) = h_{ij} \tag{18}$$

VI. Validate the following:

$$\begin{aligned} e(v,h) \prod e\left(v_{ij} + v_{i-1,ij}, h_{i-1,ij}\right) \\ \cdot e\left(v_{ij} + v_{i-1,ij}, x_{i-1,ij}\right) \\ \cdot e\left(v_{ij} + v_{i-1,ij}, y_{i-1,ij}\right) \\ = e(g, \delta_{31}) \end{aligned} \tag{19}$$

5 SAFETY

This scheme indicates that it is safe when following an attack. The original content creator and i following author can justifiably claim the copyright. The responsibility for content creation note to clarify. This satisfies in full the requirements described as follows.

Aggregate Signature Validity:
We next describe the prevention of forging of author signatures. Even a third party does not participate in the aggregate signature used to obtain all public information, which indicates that forging an aggregate signature is difficult.

Disguised Participation Prevention:
For third parties not involved in content creation, the inability to make chased responsibility for the content freely. In other words, for the signer participating in the aggregate signature, adding a third party who is not participating in the aggregate signature as a free signer is difficult.

Collusion Attack Inability:
Two people is next to a certain piece of content performing the signature creation. To escape responsibility, the two people are for contents that is created by the collusion. In other words, for the signer that participates in aggregate signature, collusion is difficult to be denied participation in aggregate signed.

5.1 Aggregate Signature Validity

Forging an aggregate signature is difficult. Consider for the following example of the aggregate signature δ_{31}, as discussed in the previous section. If counterfeiting δ_{31} is made difficult, aggregate signature validity is ensured. An Attacker A play that the random verification key to be treated as the verification key is one of the signer has the input value.

Depending on whether the hierarchy is assumed to output all of the signature and verification key pair the rest. If Attacker A is successful, to input value, aggregate corresponding to the assumed hierarchy. Attacker A can then output the signature δ_{31}. This means that a successful δ_{31} has been forged. In this case, the following theorem holds.

Theorem 1: In the random oracle model, forgery is difficult and the BLS signature forgery is equivalent to difficulty of δ_{31}.

Proof 1: A is an attacker attempting to forge δ_{31}, B is an attacker attempting to forge a BLS signature.

The fact that A attacks succeed if B's also succeed is self-evident. Thus, by showing that B attacks succeed if A's succeed, we show that both attacks are equivalent. First, Attacker B holds a verification key v_{11}. In addition, a response to the random and signature oracles is responded. For this verification key v_{11}, $s_{11}g$ can be represented and B is set to unknown value of s_{11} (Boneh, nd.).

B to run the A as Honest Player. Here and B give

$v`_{11}$ to A. A outputs a pair of verification keys: $v`_{11}$ and the other signature key $x`_{11}$. In addition, A determines $\delta`_{31}$ by the response to the random and signature oracle in order to reply to B. B use $\delta`_{31}$ in (12),

$$\begin{aligned}\delta`_{31} - s_{31}h_{31} - \delta`_{22} \\ -s_{21}h_{21} - \sum_{j=2}^{4}(\sigma_{1j} + T_{1j}) \\ -s_{21}\sum_{j=1}^{4}(x_{1j} + y_{1j}) = \sigma`_{11}\end{aligned} \quad (20)$$

Because it can be calculated, the BLS signature $\sigma`_{11}$ (Boneh, 2001) corresponding to $v`_{11}$ can be created. Thus, a B attack is successful and the aggregate signature $\delta_{31}$is proven to be safe.

5.2 Disguised Participation Prevention

Disguised participation refers to attacks added to the fraud systematically the signer. For example, a signer does not participate in signature schemes. In a scheme, aggregate signature δ_{31} contains all of the things that the participating signer indicates that dale. In other words, as a means of intervention, a free signer is considered a new δ_{41}. If the validity of this signature δ_{41} is denied, it is proved

Certification 2: A third party not participating as an attacker does not participate in the content, to make information from their signature key. Generate δ_{41} from public information, the aggregate signature to the content though because it is forged as participating. Thus indicating that the δ_{41} is inconsistent. If A is not responsible for the content, the data have not been edited. It can be seen that no information exists for A because in the control state of the editing control of Fig. 2. Consequently, that it does not exist in the control state is apparent. Thus, A is without permission and put a signature on the content, it is difficult to make responsible.

5.3 Collusion Attack Inability

A collusion attack is one that holds by shape to make responsible for the other person to escape the responsibility or virtual person. This part shows the adjacent $x_{11}(s_{11} + s_{21})$ of the aggregate signature δ_{21} of forgery is considered to be satisfactory if it can be shown that difficult.

Certification 3: To indicate that a forged hash value of content x_{ij} with signer key s_{ij} shows an adjacency relationship is acceptable. However, forging δ_{21} is difficult using Proof 1. Consider the safety of the alternative.

For a single content from the (13), it can be expressed only adjacent two adjacent relationship.

$$\delta_{21} = x_{11}(s_{11} + s_{21}) \quad (21)$$

This is represented by the product and becomes $x_{11}(s_{11} + s_{21})$. In other words, this $s_{11} + s_{21}$ is forged by collusion with the adjacent signer, and, therefore, changing to $s`_{11} + s`_{21}$ is possible. This was collusion adjacent author of $s_{11} + s_{21} = s`_{11} + s`_{21}$ it is possible to be in _(21) become such a key, virtual signer $s`_{11}, s`_{21}$, it is possible to escape as the responsibility of the content. A means to prevent this, as it is one of the signing keys for element Z_p^*. Any two sets of values of the sum to be selected are the elements providing a key space K (Erdös, 1980) such that a unique signature key becoming the element of K can be avoided. Accordingly, while lowering the safety of the key can be considered, we can safely increase the number of bits.

6 SUMMARY

Secondary use of editors within the scope of an author's intention is easy. Editing control and rights inherited notation have not been simultaneously realized using conventional methods in constructing a practical system. This proposed system showed that it is safe by using a security prevention.

By using this service, anyone who produces content may prove right. In addition, secondary users can inherit information related to the original content. Original content containing edit controls can indicate the intention of the primary author. This is the case for incorrect content as well, because the service side is willing to regulation, it has become a safe service.

REFERENCES

YouTube, https://www.youtube.com/

CLIP, http://www.clip-studio.com/clip_site/

Tetsuya, I., Makoto, S., Kunihiro N., Kazuo O., & Masahiko T, 2007. Sanitizable Signature Schemes based on Aggregate Signature. *Symposium on Cryptography and Information Security*. 2C4-3.

Tatsuhiko, S., Yoshio, K., & Masaki, I., Keiichi I, 2011. E-signature scheme that can control the edit for the contents. *Computer Security Symposium*.

Masaki, I., & Keiichi, I, 2012. An Expression of a Quotation Process in Contents with a New

Tree-structure Specified Aggregate Signature Scheme. *Information Processing Society of Japan.* Vol.53 No.9 2267–2278, Sep.

Kunihiko, F., & Yasuyuki, T., 2008. A Formal Foundation for Creative Commons Legal Codes. *Information Processing Society of Japan.* Vol.49, No.9, pp.3165–3179.

Katsuhiko, K., Katashi, N., 2006. An Annotation Platform for Meta-Content Processing. *Information science and technology Letters -FIT.* Vol.5, pp.381–384.

Boneh, D., Gentry, C., Lynn, B., & Shacham, H., 2003. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. *Advances in Cryptology – EUROCRYPT.* LNCS 2656,pp.416–432, Springer.

Masaki, I., & Toshiaki, T., 2009. Copyrigh Protection Scheme Enabling Identification of Data Composition in Secondary Use of Digital Contents. *SCIS.* 1B2-4.

Yamamoto, D. & Ogata, W., 2006. Structured Aggregate Signatures. *Symposium on Cryptographyand Information Security –SCIS.* 3A4-3.

Masaki, I., Ryu, W., & Toshiaki, T., 2010. Proposal and Evaluation of a Hierarchical Multisignature Adapted to Browsing Verification of a Document for Circulating. *The Institute of Electronics, Information and Communication Engineers.* Vol.J93-B, No.10, pp.1378–1387.

Komano, Y., Ohta, K., Shimbo, A., & Kawamura, S., 2005. On the Security of Probabilistic Multisignature Schemes and Their Optimality. *Cryptology in Malaysia –Mycrypt.* LNCS 3715, pp.132–150, Springer.

Komano, Y., Ohta, K., Shimbo, A., & Kawamura, S., 2008. Provably Secure Multisignatures in Formal Security Model and Their Optimality. *IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences.* Vol.E91-A, No.1, pp.107–118.

Helena, R., Jordi, H., 2007. An Interdomain PKIModel Based on Trust Lists. *4th European PKI Workshop: Theory and Practice, Euro PKI.* Palma de Mallorca, Spain, June 28-30.

Boneh, D., & Franklin, M. K. nd, Identity-Based Encryption from the Weil Pairing. *Advances in Cryptology –CRYPTO.* LNCS 2139, pp.213–229, Springer.

Boneh, D., Lynn, B., & Shacham, H., 2001. Short Signatures from the Weil Pairing. *Advances in Cryptology –ASIACRYPT.* LNCS 2248, pp.514–532, Springer.

Erdös, P., 1980. Some Applications of Ramsey's Theorem to Additive Number Theory. Europ. J. Co.