社団法人　情報処理学会　研究報告
IPSJ SIG Technical Reports

2005－ＣＳＥＣ－30 (29)
2005／7／21

Word 2003 XML 文書への情報ハイディングシステム

北野　宗之†　　増田　英孝†　　中川　裕志††

† 東京電機大学工学部　〒101-8457 東京都千代田区神田錦町 2-2
†† 東京大学情報基盤センター　〒113-0033 東京都文京区本郷 7-3-1
E-mail: †kitano@cdl.im.dendai.ac.jp, †masuda@im.dendai.ac.jp, ††nakagawa@dl.itc.u-tokyo.ac.jp

あらまし　本研究は，情報ハイディングとデジタル署名を組み合わせて，文書作成者が保護したい部分に対してデジタル署名をかけ，その署名値を第三者に気付かれないように文書中に埋め込んで受け手に渡し，受け手が保護対象部分の改竄の有無を検証することができる手法について検討している．本稿では，現在広く利用されているワードプロセッサ Word 2003 で出力可能な XML 形式のファイルを対象とし，情報を埋め込んだ文書を Word 2003 上で表示しても情報を埋め込んでいることが発見されにくいように XML 要素の色情報を利用した手法を提案する．この方法を利用し，Word 2003 上で指定した保護部分の改竄の有無を検出できるシステムを試作した．
キーワード　情報ハイディング，XML，Word 2003，デジタル署名，改竄検出，色属性

An Information Hiding System for Word 2003 XML Documents

Muneyuki KITANO†, Hidetaka MASUDA†, and Hiroshi NAKAGAWA††

† School of Engineering, Tokyo Denki University　Kandanishiki-cho 2-2, Chiyoda, Tokyo, 101-8457 Japan
†† Information Technology Center, University of Tokyo　Hongo 7-3-1, Bunkyo, Tokyo, 113-0033 Japan
E-mail: †kitano@cdl.im.dendai.ac.jp, †masuda@im.dendai.ac.jp, ††nakagawa@dl.itc.u-tokyo.ac.jp

Abstract　We are considering an information hiding technique for the following situation: 1) The original writer applies the digital signature to the part of the document where she/he wants to protect. 2) The signature will be embedded in the entire document itself without being noticed by the relayed third party. 3) After the final recipient received it, she/he can detect whether the protected portion is tampered or not. Our target is XML format document which can be output by the widely used editor: Microsoft Word 2003. We propose technique which is hard for third party to perceive whether some data are embedded by an information hiding system with modifying color attribute of XML element, even if the document with the embedded information was displayed on Word 2003. We implemented a prototype system which embedes the hidden information and detects whether tampered or not.
Key words　Information Hiding, XML, Word 2003, Digital Signature, Tamper Detection, Color Attribute

1. まえがき

近年，ネットワークの普及により，文書を XML 形式で電子的に交換することが増えている．XML は，文書構造を表現するための汎用的なメタ言語であり，Web コンテンツや電子商取引などの様々な文書の記述や交換に広く利用されている．それに伴い，XML 文書のセキュリティ確保の必要性が高まっている．

そこで，情報の存在自体を第三者に気づかれないようにする技術として情報ハイディングが注目されている．情報ハイディングは，埋め込み元のデータに対して，第三者に分からないように情報を隠し，また隠した情報を必要に応じて抽出することを目的とし，電子的コンテンツの著作権保護 (電子透かし) や秘匿通信路 (ステガノグラフィ) の基盤技術として位置づけられる．

これまで文書を対象とした情報ハイディング手法のアプローチの一つとして，行間，文字間隔を調整するなど文書自体を画像として取り扱う手法としてホワイトスペース法 [1]，ヌルキャラクタなどの文字として表示されないキャラクタを挿入することにより，埋め込みを実現する手法として文字埋め込み法 [2] などがある．

次に，文書内容そのものはプレインテキストとして扱い，同義語等を利用して文書内容を書き換える方式があり，辞書変換法と呼ばれる [3], [4]．言い換え辞書によらないテキストの意味的情報量を利用した手法もある [5]．

また，テキスト中の改行位置の変更による情報埋め込み手法

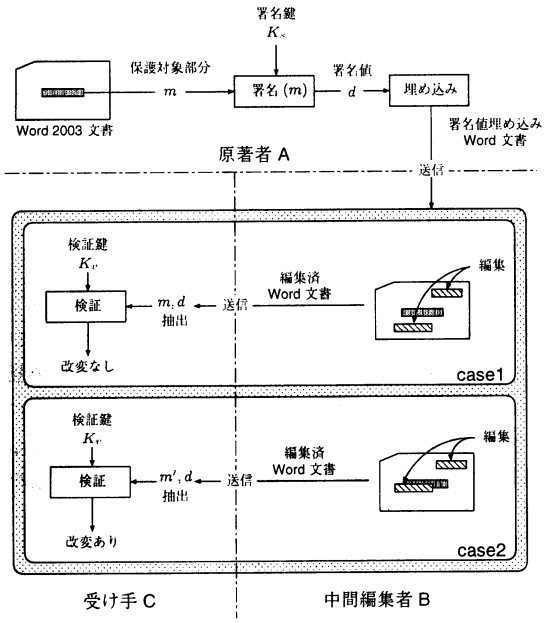


図 1　ビジネスモデルの概念図

Fig. 1　A concept of business model

もある [6]．本研究では半構造を持った XML 文書を対象として秘匿情報を埋め込むことを目的としている．

一方，改竄やなりすましを防ぐ方法としては，デジタル署名が信頼された手法として広く利用されている．デジタル署名は，そのメッセージが確かに文書作成者である利用者から送られてきたものかどうかを証明できる「認証」と，文書作成者が第三者に対し，そのメッセージを送信したことを否定できない「否認不可」を実現するための技術として広く一般的に用いられている．しかし，デジタル署名では暗号化された署名データが存在することが明示的にわかるため，攻撃の対象とされてしまう危険性がある．また，文書全体を署名対象とするよりもある部分のみを保護したいという要求もある．

そこで我々は，XML 文書に対するデジタル署名と情報ハイディングを組み合わせた著作権保護技術として，現在広く利用されているワードプロセッサ Word 2003 で出力可能な XML 形式のファイルを対象とし，情報を埋め込んだ文書を Word 2003 上で表示しても情報を埋め込んでいることが発見されにくい手法を提案する．提案手法は，空白及びタブ文字の色情報が変化しても，Word 上の表示には影響を与えないことを利用して情報を埋め込み，受け取り側で送り手側が指定した保護部分の改竄の有無を検出できる情報ハイディング手法であり，この方法によって試作したシステムについても報告する．

2.　本提案の目的と応用

本研究では，文書全体をデジタル署名の保護対象とするのではなく，著者が指定したある特定の部分のみを保護することを目的とする．最終的な文書の受け手は原著者の保護対象部分に改竄があったかどうかを検証することができる．原著者と最終的な受け手の間に第三者が介入し，保護部分以外の編集は許すものとする．

```
<w:body>
 <w:p>
  <w:r>
   <w:rPr>
    <w:rFonts w:hint="fareast">
    <w:color w:val="000000">
   </w:rPr>
   <w:t>Tokyo Denki University</w:t>
  <w:r>
 </w:p>
<w:body>
```

図 2　Word 2003 XML 文書の例

Fig. 2　An example of Word 2003 XML document

次に，提案する情報ハイディングシステムを利用したビジネスモデルについて説明する．図 1 に我々が提案する情報ハイディングシステムにおけるビジネスモデルの例を示す．文書を作成する原著者 A は，文書の中の指定した部分 (網かけ部) を保護対象 m とし，その部分に署名鍵 K_s を用いてデジタル署名を行い，署名値 d を文書内に分散させて埋め込む．この保護対象部分 m に関しては第三者からの加筆や訂正などの編集行為は許さないものとする．この署名付き文書を中間編集者 B に送信し，中間編集者は原著者が指定した保護対象部分以外の箇所においては編集を許されている．図 1 では，中間編集者が保護対象部分以外の編集を行った場合を case1，保護対象部分に編集を行った場合を case2 として示す．そして，最終的な文書が受け手 C に送信され，C は中間編集者 B によって原著者 A の保護対象部分に変更が加えられていないことを検証鍵 K_v を用いて確かめることができる．

さらに，この文書全体を M とおくと，中間編集者は原著者の署名とは独立に M の署名又はハッシュを埋め込み，最終的な文書として受け手に渡すこともできる．

図 1 のシナリオでは，case1 の編集済文書では m の改変はない，case2 の編集済文書では保護対象部分 m' は改変されていると判定される．

このモデルの応用として，社内で取りまとめるマニュアルやドキュメントを一人ではなくグループまたは組織で作り上げようとする場合などの利用，原作者が書いた意見やデータを中間編集者が歪めたことを検出するシステムである．

3.　情報ハイディングとデジタル署名

3.1　Word 2003 XML 文書

本研究の対象である Word 2003 XML 文書について概要を説明する．広く利用されている Microsoft Word 2003 では，独自のバイナリ形式 (DOC) の他に XML 形式でファイルを入出力できる．本稿では，以後，*XML 文書* は XML 形式でファイル保存した際の Word 2003 XML 文書 を示すものとする．

XML 文書の階層構造及び要素や属性に関する適用規則は，

Microsoft が規定した Office XML スキーマ [7], [8] により定義されている．Word 2003 で用いられる Office XML スキーマは WordProcessingML と呼ばれている．

WordProcessingML に準拠した XML 構造は，wordDocument 要素をルート要素とし，その子要素には，ドキュメントプロパティ，フォント，スタイルなどを定義する要素と，その文書の本体を記述する要素からなる．

文書本体は body 要素と呼ばれ，body 要素の構造は図 2 のようになる．ここで各要素の接頭辞 w は WordProcessingML スキーマ群の名前空間の 1 つであり，すなわち <w:body> タグは名前空間が w の body 要素である．body 要素の子要素には段落を記述する <w:p> タグ (pharagraph 要素) を持ち，さらに pharagraph 要素の子要素には文を記述する <w:r> タグ (run 要素) を持っている．run 要素はフォント情報，色情報などテキストの各種設定を <w:rPr> タグ (runProperties 要素) の子要素で制御し，<w:t> タグ (text 要素) で記述されるテキストデータに反映される．

3.2 XML 文書への情報ハイディングの目標

ここでは XML 文書に適した情報ハイディング手法を検討する．その際，埋め込まれた情報が Word 2003 上の表示で明らかに情報を埋め込んでいることが分からないようにデジタル署名などの秘匿情報を埋め込む．また，埋め込み後，Word を通して編集し上書き保存した際に，埋め込まれる情報は WordProcessingML によって失われないことを目標とする．

XML 文書への情報ハイディング手法には先に示したような同義語の置換を用いた手法や改行位置の制御による手法も提案 [6], [9] されているが，XML の論理構造に着目した手法もある [10].

そこで我々は，提案されている手法も含めて次のような埋め込み手法について可能性を検証した．

（1） 無意味な空要素タグもしくは無意味な属性を XML 文書内に付加

（2） 木構造において同階層に位置する異名要素の出現順序の入れ換え

（3） 要素内に属性が複数ある場合の属性の出現順序の入れ換え

（4） タグの “>” の前に空白を付加

（5） 空要素タグの表記の切り替え

（6） 既存の属性の値の変更

しかし，(1)〜(5) までの手法は Word 2003 でファイルを読み書きする際に，Office XML スキーマ [7], [8] によって，当然のことながら自動修正されてしまい，情報埋め込み手法としては利用できない．

(6) の方法のうち，属性値が決められた型や範囲に違反しているとその属性は削除されてしまう．

したがって，本研究では属性値の型はそのままで，かつ許される範囲の値のみを用いて情報を埋め込む方法を探さなければならない．

3.3 XML デジタル署名

XML Digital Signature 標準 [11] で定められている XML デジタル署名では，署名アルゴリズム，証明書や署名のタグを定めて，任意のデータに対する署名や XML 文書中の指定した要素や内容に対しても署名を付けることができる．

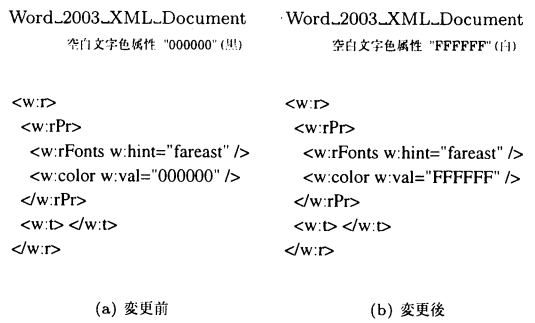


図 3 空白文字の色属性の変更
Fig. 3 Modification of color attribute of blank characters

保護対象のダイジェスト値を求め，そのダイジェスト値を子要素として XML 署名情報要素に入れる．さらにその署名情報要素に対してダイジェスト値と署名値を計算し挿入する．XML デジタル署名は XML 文書中に Signature 要素として記述される．

XML 署名には，

（1） Detached 署名: 署名要素の外部にある XML 文書または XML 以外のデータを署名対象とする，

（2） Enveloped 署名: 署名文書要素の内部に署名要素を含む，

（3） Enveloping 署名: 署名要素の内部に署名対象要素を包含する

の 3 つの形式がある．(1) は外部ファイルの署名に用いられ，(2) は署名文書に署名を付ける場合に用いられる．(3) は独立した署名要素を他に引き渡したりするなどの用途に用いられる．(1) の形式では外部ファイル全体が保護対象となり，その中の一部を保護対象としたい場合には利用できない．(2) と (3) の形式では，XML 文書中に署名情報が要素として埋め込まれるため，特定のタグを検索することにより，署名が埋め込まれていることがわかってしまう．

本研究では，文書内の一部の保護対象部分にデジタル署名を適用し，その署名情報を XML 文書中に署名が明らかに埋め込まれているようには見せないことを目的とするため，(1)〜(3) の形式をそのまま利用することはできない．

4. Word XML 文書の色属性を利用した情報ハイディングの提案

4.1 情報埋め込み手法

XML 文書中の要素内にある属性のうち，その変更が Word 上の表示には影響を与えないものを選択して情報を埋め込む．情報の埋め込みにはできるだけ一度に埋め込むことができる情報量が多いことが望まれる．そこで本研究では XML 要素の色

表 1　日本語 Word 文書の空白数の調査

Table 1　The result of counting the number of blanks on Japanese Word documents

全文書数	115
平均空白数	241.4
平均ページ数	2.4
1 ページ当たりの空白数	111.5

属性に着目した．空白，タブ文字，段落自体を利用すれば，色情報を変更しても Word 上の見かけに影響を与えないことを利用する．

例として，半角の空白文字の色属性を “000000”(黒) から “FFFFFF”(白) に変更した場合の，XML 文書及び Word 上の表示を 図 3 に示す．図 3(a)，図 3(b) に示した通り，XML 文書をテキストエディタ等を用いて変更した要素を調べると色属性に変更が加えられたことがわかる．しかし，XML ソース全体を俯瞰した程度では埋め込み情報が目立たない．次に，Word 上の表示においては空白文字の色属性の変更による影響はない．この色属性を利用すれば，空白 1 文字当たり 24 ビットの情報を埋め込むことができることになる．

Word では編集記号の表示モードを変更することにより空白文字を “·”，タブ文字を “→” として表示することができる．しかし，空白及びタブ文字の色属性がどんな色であっても “·” 及び “→” の表示は色属性に関係なく一定色であり，色属性の変更が Word 上の表示に影響を及ぼさない(注1)．

空白が多数出現する英語などの言語では、この方法は多数の埋め込み位置を使えるので有力である．これにひきかえ，日本語文書では空白文字の出現頻度が低い．そこで，空白およびタブ文字が実際の Word 文書中にどれくらい出現するのか，筆者以外の第三者が作成した学内外の 115 文書について Word の文字数カウントツールを用いて調査を行った．その結果を表 1 に示す．この表中の空白数は半角の空白，全角の空白，タブ文字の合計数である．これらの空白文字は，主として体裁を整えたり，表示を微調整するために使われていた．後に 5.1.3 で述べるようにデジタル署名値の埋め込みには 64 個の空白が必要である．従って，表 1 の結果からすれば多くの文書では署名値の埋め込みに十分な空白が存在する．

4.2　保護対象部分の指定

受け取り側で保護対象部分の改竄を検証するためには，保護対象部分の位置を示す情報 (以後，チェックマークと呼ぶ) を付加する必要がある．ただし，このチェックマークも Word 上の表示には影響ができるだけ少ないものとする．

そこで，画面に表示される文字の色属性を 1bit 程度変更しても，一般的には利用者が認識できないことを利用して，保護対象部分の前後の位置の要素にチェックマーク情報を付加する．まずここでは，保護対象の前後の文字が文書全体に反映されているデフォルト色属性で記述されているものとして説明する．受け取り側では，検証時にデフォルト色属性からの文字色の 1bit の差を見つけて，チェックマークに囲まれた部分が保護対象部分であることを知ることができる．

(注1)：Word 2003 の互換ソフトウェアである OpenOffice では空白およびタブ文字の編集記号に色が反映されてしまう．

表 2　Word における色属性の表記

Table 2　Representation of color attributes on Word

デフォルト属性	style 要素内の色要素	本文中の色要素
RRGGBB	ある	なし
auto	なし	なし

表 3　チェックマークの仕様

Table 3　Specifications of a check mark

R の下位 1bit	保護対象部分の先頭を示す
G の下位 1bit	保護対象部分の末尾を示す
B の下位 1bit	チェックマークであることを示す

Word で指定できる色属性の値は Red, Green, Blue がそれぞれ 8bit からなる 16 進の “RRGGBB” の 24bit 値 又は “auto”(自動) である．通常，Word のデフォルトの既定値は “auto” であるが，前景色の設定を変更して任意の色とした場合，その色属性は <w:style> タグ (以下 style 要素) 内の文字スタイルで定義される．

そこで，デフォルト色属性を “RRGGBB”，“auto” にした場合の style 要素で定義される色要素の有無と，本文中の文，段落要素に記述される色要素の有無を示したのが表 2 である．本文中でデフォルト色属性が使用されている場合には，色要素 <w:color> タグが表れないため，style 要素から色属性が何かを見つけなければならない．

表 2 より，デフォルト色属性が “auto” 以外の “RRGGBB” ならば，style 要素で定義されているデフォルト色属性を抽出して，表 3 に示した方法でチェックマークを付加することができる．例えば，デフォルト色属性が “000000”(黒) の場合，先頭を示すチェックマークは “010001”，末尾を示すチェックマークは “000101” となる．

一方，style 要素内に色要素が定義されていない場合，デフォルト色属性は “auto” であることが分かる．“auto” は黒色と見なすことができるため，“000000” として取り扱う．

4.3　情報埋め込み

ここでは XML 文書の送り手が秘匿情報を文書内に埋め込む手法を示す．図 4 にその概要を示し，その流れは以下の通りである．埋め込むデジタル署名値の生成および署名値の検証については，一般的に利用されているデジタル署名の方式をとった．

- Word 2003 上で文書作成者であるユーザ (原著者) は，保護対象部分を指定した Word 文書を XML 文書として出力する
- チェックマークを基に，指定された保護対象部分のテキスト m を抜き出す
- ハッシュ関数 h により，テキスト m のハッシュ値 $h(m)$ を計算する
- 公開鍵暗号により，ユーザの秘密鍵 S_K を使ってデジタル署名 $S_K(h(m))$ を生成し，ここではこれを d とおく
- 署名値 d を空白文字の色属性に分散させて埋め込む

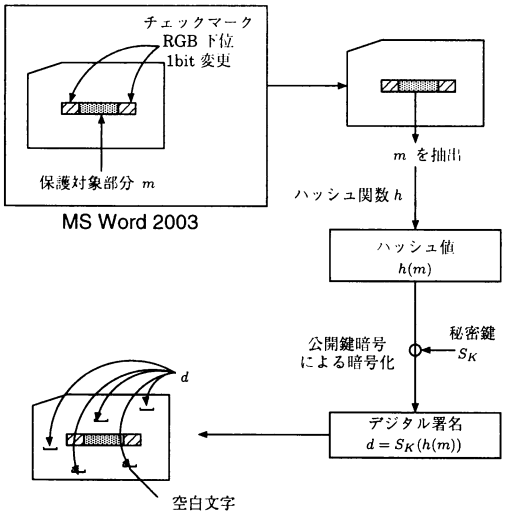


図 4 情報埋め込み方法

Fig. 4 A method of information hiding

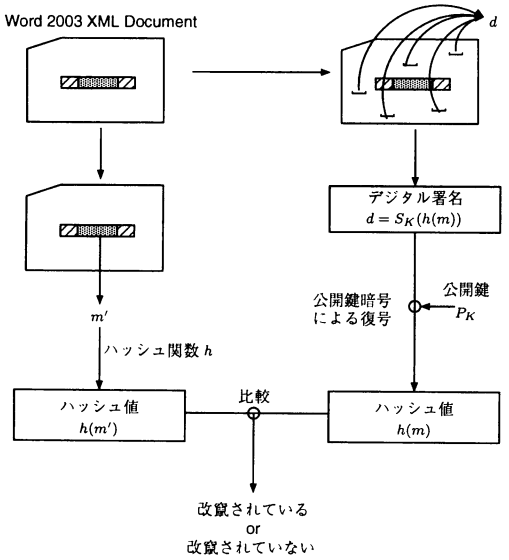


図 5 情報抽出と検証方法

Fig. 5 A method of information extraction and verification

ところで，2. 節のビジネスモデルで述べたように中間編集者は保護対象以外の部分の編集は行うことを想定している．ところが，署名値は保護対象部分以外の部分に表れる空白文字に埋め込まれる．従って，署名値の埋め込まれた部分が編集されてしまう可能性がある．ここでは，中間編集者の行う編集において元の文章が部分的に削除されることはないという場合を扱う．その場合は，署名値が埋め込まれた空白文字の出現順序が入れ換わる可能性がある．その場合でも署名値が復元できるように署名値を埋め込む各空白文字にシーケンス番号を付けておくことにする．

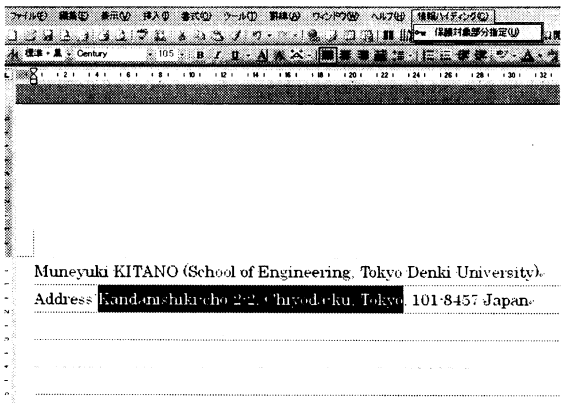


図 6 保護対象部分指定 GUI

Fig. 6 GUI to specify the region of target to protect

ここで，空白文字の色属性には 4.2 のチェックマークに入る可能性があるため，保護対象中及び前後に空白文字が存在する場合には署名値の埋め込みの対象外とする．

一方，署名値が埋め込まれた空白文字が削除される場合まで考えるとなるとデジタル署名を埋め込むことは困難である．その場合にはむしろハッシュ値を直接埋め込む方法が考えられる．ハッシュ値が埋め込まれた空白が部分的に削除されたとしても残りの部分のハッシュ値が保護対象部分から生成したハッシュ値と一致しているならば改竄されていない可能性の高いことが推定できる．ただし，埋め込み方法が攻撃者に見破られた場合には，ハッシュ値自体を改竄して埋め込むことによって改竄されていないかの如く偽装することが可能となる．この考え方の実現方法と改竄に対する耐性については今後の検討課題である．

4.4 情報抽出と検証

ここでは情報埋め込みが行われた XML 文書に対して，受け取り側で秘匿情報を抽出して改竄されているかどうかを検証する．図 5 にその概要を示す．

- 空白文字に埋め込まれた色属性から d を復元する
- 公開鍵暗号より，公開鍵 P_K を使って $d = S_K(h(m))$ を復号し $h(m)$ を得る
- チェックマークを基に，保護対象部分のテキスト m' を抜き出す
- ハッシュ関数 h により，m' のハッシュ値 $h(m')$ を求める
- $h(m)$ と $h(m')$ を比較し，改竄があるかないかを判定

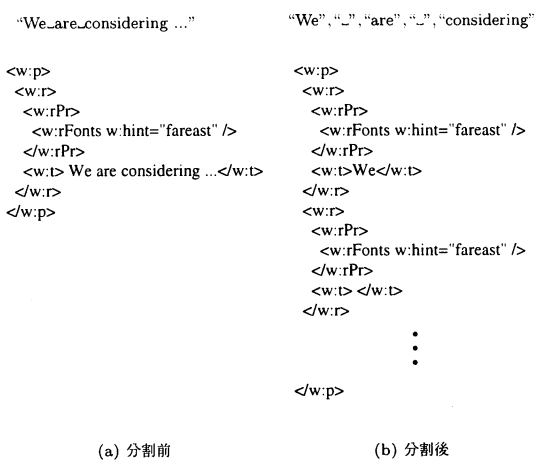
以上の工程により，保護対象部分 m について改竄の有無を検出することができる．

5. Word XML 文書への電子署名の埋め込みと検証

ここでは，4. 節の情報埋め込み手法についての具体像を示す．情報埋め込みは，保護対象部分の指定方法，受け取り側の保護対象部分の特定手段としてのチェックマーク付加，埋め込みに用いる署名値と埋め込みについて説明する．情報抽出と検証は，保護対象部分の抽出方法，空白文字に埋め込まれた署名

図 7 チェックマーク付加後の XML 文書 (デフォルト色属性 =“auto”)

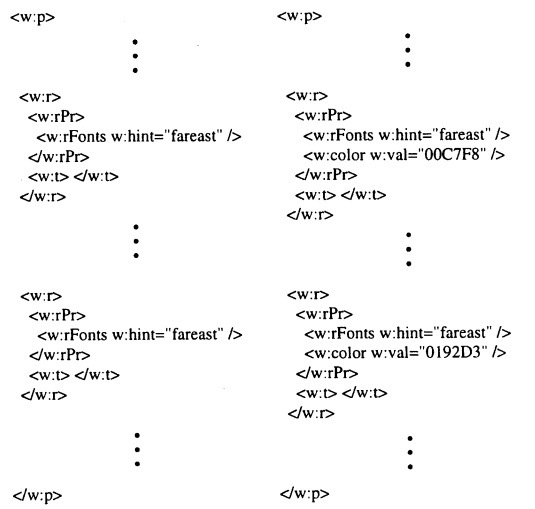
Fig. 7 An example of XML document with check marks (default color value is equal to “auto”)



(a) 分割前 (b) 分割後

図 8 空白文字要素への分割

Fig. 8 Division of the text into the content and each blank for embedding



(a) 情報埋め込み前 (b) 情報埋め込み後

図 9 情報埋め込みの結果

Fig. 9 Example of XML color values after hiding

値の復元，両者の署名値の比較について説明する．

5.1 情報埋め込み

本設計は，4.3 の情報埋め込みを XML 文書を解析する DOM [12] の木構造を利用して実現する．以下に要件を示す．

5.1.1 保護対象部分の指定

4.3 の保護対象部分を指定するには，利用者の立場からすると専用の GUI を用意することが望ましい．今回は，Word のプラグインとして実現することを試みた．GUI は Word の VBA を用い，保護したい部分を選択して反転表示させ，メニューバーから保護対象部分指定を選択するだけで，保護対象部分の指定を行うことができる．GUI の使用例を図 6 に示す．

5.1.2 チェックマーク付加

文書作成者が指定した保護対象部分に対して，4.2 に示したように保護対象部分前後の XML 要素の色属性に変更を加える．ここでは図 6 について，Word 文書のデフォルト色属性を “auto” とした場合の，チェックマーク付加の例を図 7 に示す．

5.1.2 より，“auto” を “000000”(黒) として処理する．保護対象部分の直前のテキスト “:” の要素の色属性には，先頭を示すチェックマークとして “RRGGBB” の R, B の下位 1bit を変更した，“010001” が設定される．直後のテキスト “,” の要素の色属性には，末尾を示すチェックマークとして “RRGGBB” の G, B の下位 1bit が変更された “000101” となる．

5.1.3 デジタル署名値の埋め込み

今回の実装では，4.3 のデジタル署名のアルゴリズムとして SHA1withDSA を用いることにした．生成されるデジタル署名値は 1024bit となる．また署名，検証を行うための秘密鍵，公開鍵の生成には，Java の keytool [13] を利用している．色属性値は，空白文字の色属性 24bit を シーケンス番号 ($8bit$) + 署名データ ($16bit$) に分割して利用する．これにより，デジタル署名の 1024bit をすべてを埋め込むために空白文字が 64 個必要となる．ここでは，保護対象部分のテキストのデジタル署名値を計算した後，1 番目の空白文字から順にこれらの値を埋め込んでいく．

ところで，署名値を色属性に埋め込むにあたり，1 個の空白文字は 1 つの run 要素として単独で構成されている必要がある．しかし，空白文字に隣接するテキストが同じフォント情報，色情報を持っていれば，1 つのテキストデータを持つ 1 個の run 要素として扱われてしまうために空白文字の色属性だけに情報を埋め込むことができない．このような場合，空白文字 1 個が単独の run 要素となるように XML の構造を情報埋め込み処

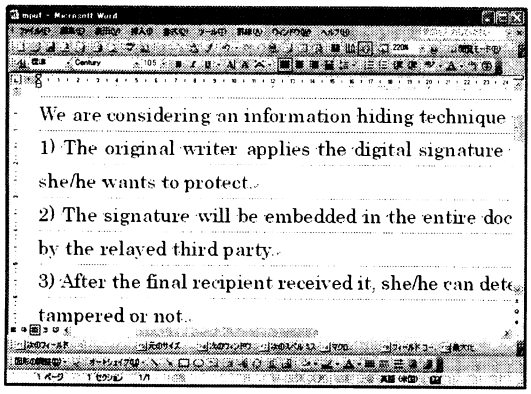


図 10　埋め込み前の Word 画面表示例

Fig. 10　An example of displaying a document on Word before embedding

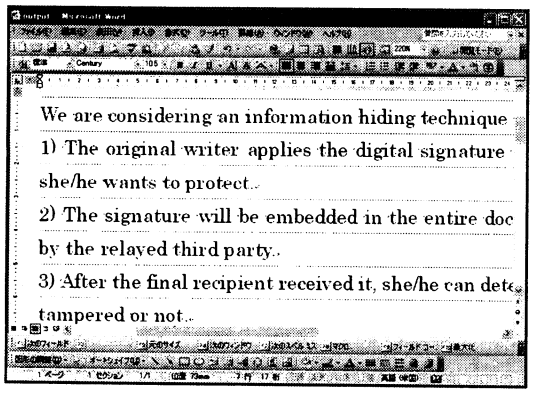


図 11　埋め込み後の Word 画面表示例

Fig. 11　An example of displaying a document on Word after embedding

理を行う前に分割しておく必要がある．この例では，図 8(a) 示す “We␣are␣considering ...” というテキストを持った要素をそれぞれ図 8(b) に示す “We”, “␣”, “are”, “␣”, “considering” のように空白とそれ以外に区切って要素とする．

5.1.1 より抽出したテキスト m に対する情報埋め込みの実行例を，図 9 に示す．デフォルト設定で記述されている文要素タグ <w:r> は，図 9(a) のように色要素タグ <w:color> を持っていないため，ここでは色情報を持たない空白文字に対して <w:color> を新たに作り，図 9(b) のように空白文字の色属性 “C7F892”,”D3DFF8” を付加する．これらの値は，分散して埋め込む情報の一部である．デジタル署名を埋め込んだ後に，Word XML 文書に残存してしまう VBA スクリプトを消去しておく．

図 10 に埋め込み前の文書，図 11 に埋め込み後の文書を示す．この 2 つの文書に Word 表示上の違いは見られないため表面的に見ただけでは情報が埋め込まれていることが発見されにくい．

5.2　情報抽出と検証の設計

ここでは 4.4 の情報抽出と検証について説明する．2 の保護対象部分のテキストを抜き出すために，5.1.2 のチェックマークの色情報を利用する．また，空白文字の色属性に埋め込まれた署名値の復元は，1 番目の空白文字から順にシーケンス番号を確認しながら抽出していく．

表 4　文書編集による検証結果

Table 4　The results of verification by editing the document

評価項目	Word 編集	エディタ編集
保護対象部分のテキストを変更	×	×
空白文字の色属性を変更	×	×
情報を埋め込んだ空白文字を削除	×	×
別の色つき空白文字を挿入	×	×
関係のない部分を削除以外に変更	○	○

図 11 の XML 文書に対して，以下の項目について Word 上での編集とテキストエディタ上での編集を行った場合について，検証を行った．

（1）　保護対象部分のテキストを変更する
（2）　空白文字に埋め込んだ色情報の一部を変える
（3）　情報を埋め込んである空白文字を削除する
（4）　色属性を指定した空白文字を挿入する
（5）　保護対象部分とは関係のない部分を削除以外の方法で変更する

項目の (1)〜(3) は保護対象部分に関わる改竄であるため，検証結果は不一致 (×) を返すことを想定し，(4) ではシーケンス番号を崩してしまう値が使われている場合には不一致 (×) となる．(5) では一致 (○) を返すことを想定している．本システムでの検証結果は表 4 のようになり，我々が想定していたものと一致している．このうち (2) については，文書の空白に余裕があれば同一の署名値を繰り返し埋め込むことで，局所的な色属性変更による耐性は持たせることができる．

今回は検証システムを受け取り側で持っている場合を想定しており，その場合に情報埋め込みが行われた XML 文書を受け取ることにより，受け取り側で保護対象部分の改竄の有無を検証できる．

6.　おわりに

Word 2003 でファイル入出力可能な XML 文書に対して，保護対象部分を指定してそのデジタル署名を秘匿情報として埋め込む手法を提案した．提案手法では，空白及びタブ文字の色属性を利用して情報を埋め込み，受け取り側で情報を抽出して改竄の有無を検証できる．

今回は Word 2003 で作成された XML 文書について，提案手法を用いたシステムの試作を行った．その結果，保護対象部分の埋め込み，抽出後の検証が行えることがわかった．また，保護対象部分の改竄の有無の検出実験を行った結果，あらかじめ期待された結果が得られた．

本手法の秘匿情報の埋め込み対象は，元の Word 文書の本文に表れる部分についてのみに言及しているが，ファイル名や変更履歴などのドキュメントプロパティを埋め込むことも考えられる．あるいは文書とは全く関係のない通信者間だけが秘密に

情報を送受するといった利用もある．

今後は，Word 2003 に情報埋め込み検証システムを組み込んで，すべての作業を利用者が Word 2003 上で一括して処理が行えるようにする予定である．これが可能になれば，XML 形式の文書だけではなく，Word 2003 の DOC 形式のままですべての工程を扱うことができる．

また本稿では，1 文書に対して 1 個の保護対象部分への埋め込み及び検証を行った．しかし，1 文書中に複数箇所の保護対象部分を指定したい場合には，それらを繋げて 1 つの保護対象として捉えた署名の埋め込みも検討している．

その他，本手法は情報が埋め込まれた文書が元文書に比べて色情報を多く含んでいるために，ファイルサイズが増加する．このファイルサイズの増分が，その文書に情報が埋め込まれていることを第三者に安易に気づかせてしまうものかについての検討も行う必要がある．

謝　辞

本研究を進めるにあたって有益なアドバイスをして頂いた 横浜国立大学 松本勉教授，三菱総合研究所 村瀬一郎氏，井上信吾氏，赤井健一郎氏，牧野京子氏，独立行政法人情報通信研究機構 滝澤修氏，吉岡克成氏 に感謝致します．

文　献

[1] J. Brassil, S. Low, N.F. Maxemchuk and L. O'Gorman, "Electronic Marking and Identification Techniques to Discourage Document Copying," IEEE Journal on Selected Areas in Communications, vol.13, no.8, pp.1495–1504, 1995.

[2] S. Low, N.F. Maxemchuk, J. Brassil, and L. O'Gorman, "Document Marking and Identification using Both Line and Word Shifting, " Infocom '95, vol.2, pp.853–860, Boston, Massachusetts, USA, Apr.1995.

[3] M. Chapman, G. Davida, "Hiding the Hidden: A Software System for Concealing Ciphertext as Innocuous Text, " Information and Communication Security, First International Conference, pp.335–345, Beijing, China, Nov.1997

[4] 中川裕志，三瓶光司，松本勉，柏木健志，川口修司，牧野京子，村瀬一郎，"意味保存型の情報ハイディング–日本語文書への応用–，" 情報処理学会論文誌，vol.42，no.9，pp.2339–2350，Sept.2001.

[5] 新見道治，峯脇さやか，野田秀樹，河口英二，"SD 式意味モデルを利用したテキストベースステガノグラフィ，" 情報処理学会論文誌，vol.44，no.8，pp.1894–1903，Aug.2003.

[6] 滝澤修，山村明弘，中川裕志，松本勉，村瀬一郎，牧野京子，井上信吾，大野浩之 "改行位置の制御によるテキストステガノグラフィの提案，" 言語処理学会第 7 回年次大会，C2-4，pp.135–138，Mar.2001.

[7] E. Lentz, M. McRae, S. St.Laurent, Office 2003 XML, O'Reilly, 2004.

[8] Microsoft Office 2003 XML Reference Schemas, http://www.microsoft.com/office/xml/.

[9] C.D. Jensen, "Fingerprinting Text in Logical Markup Languages, " Information Security, 4th Iternational Conference, ISC 2001, Lecture Notes in Computer Science 2200 Springer 2001, pp.433–445, Malaga, Spain, Oct.2001,

[10] 井上信吾，村瀬一郎，滝澤修，松本勉，中川裕志，"XML におけるステガノグラフィ手法の提案，" 暗号と情報セキュリティシンポジウム，SCIS2002，pp.301–306，Jan.2002.

[11] M. Bartel, J. Boyer, B. Fox, E. Simon, XML-Signature Syntax and Processing, http://www.w3.org/TR/xmldsig-core, W3C Recommendation, Feb.2002.

[12] L. Wood et al, Document Object Model, World Wide Web Consortium(W3C), http://www.w3.org/DOM/, Jan.2005.

[13] keytool –Key and Certificate Management Tool, Sun Microsystems, Java 2 SDK, Standard Edition Documentation Version 1.4.2.