1E-03

ハッシュを用いたファイル自己証明方式の提案

山田　隆行†

高知工業高等専門学校†

1．はじめに

近年，不正アクセスによるホームページやファイルの改ざんが問題となっている[1,2]．自己の真正性を証明するためにハッシュ関数を用いる手法があるが，ハッシュ値を別途送付または掲載するなどの必要がある．

本論文では，ハッシュ値をファイルに埋め込むことで本文だけで自己の真正性を証明することが可能となる自己証明方式を提案する．

2．ハッシュによる改ざん検知

ハッシュを用いた一般的な改ざん検知を下記に示す．

(1) 送付ファイルの改ざん検知

送付ファイルの改ざん検知のしくみを図 1 に示す．ハッシュ値を比較することにより送路上でファイルが改ざんされていないことを証明することができるが，ハッシュ値を作成ファイルと合わせてハッシュ値を受信者に送付する必要がある[3]．

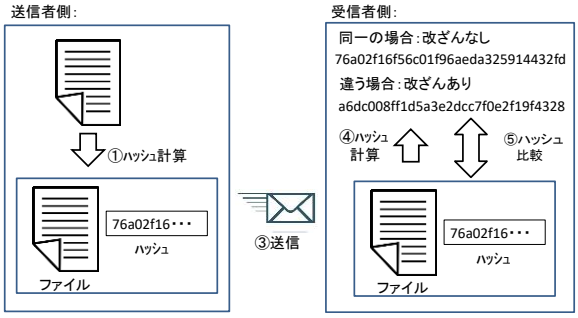

図 1　送付ファイルの改ざん検知

(2) ダウンロードファイルの改ざん検知

ダウンロードファイルの改ざん検知のしくみを図 2 に示す．ダウンロードしたファイルが，Web サイト上で配布されているファイルと同一かを調べたい場合，一般的にはチェックサムやハッシュ値などの値を計算して比較するが，ファイルのチェックサムやハッシュ値などの情報も同時に掲載する必要がある[4]．

上記のように一般的な改ざん検知では，ハッシュ値を別途送付または掲載するなどの必要があり，この取り扱いが煩雑であり，また，ハッシュ値の管理も必要となる．

本論文では，次節に述べるドキュメントのカスタムプロパティにハッシュ値を設定することにより，本文だけで自己の真正性を証明する自己証明方式を提案する．

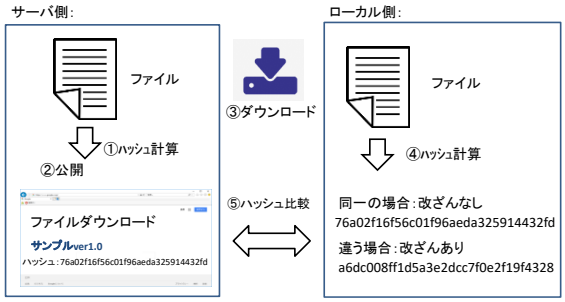

図 2　ダウンロードファイルの改ざん検知

3．ハッシュを用いた自己証明方式

3.1 ファイル形式とカスタムプロパティ

インターネット上にはさまざまなファイル形式があるが，本論文では，一般的なドキュメント(docx, xlsx, pptx, pdf)を対象とした．これらのドキュメントには，タイトルや作成者の情報を設定しておく一般的なプロパティの他に，ユーザ独自の設定情報（例えば，版の番号や編集部署など）をドキュメントに設定しておくことができるカスタムプロパティがあるが，ファイルのハッシュを用いるとハッシュ値を埋め込むときにファイル自体が改変してしまう．そこで，本文のテキストのハッシュをとり，このハッシュ値をカスタムプロパティに設定することによりファイルにテキストのハッシュ値を埋め込む．辞書攻撃やレインボー攻撃への対策として，テキストから作成することができる文字数などの付加情報や使用するソフトウエアのシリアルなどを salt としてテキストに付加することで耐性を向上させることができる[5]．

3.2 ハッシュ値の改ざんへの対策

ハッシュ関数のアルゴリズムは公開のため，悪意のある者が文書を改ざん後にハッシュ値も再計算してセットすれば改ざんを検知できなくなるという問題[5]やハッシュ値そのものが削除されてしまうことも考えられる．前者に対しては，公開鍵暗号を用いた電子署名を行う．ハッシュ値を秘密鍵で暗号化して埋め込むことにより，秘密鍵を所有する本人のみしか正しいハッ

Proposal of self-certification method for electronic documents using hashes
† Takayuki Yamada. National Institute of Technology, Kochi College

シュ値を計算できずセットすることはできない．後者に対しては，プロパティが削除されていた場合には改ざんがあったとみなすことで対応可能である．または，ファイル自体のアクセス権を適切に設定することでプロパティへの不正な書き込みや削除を制御して保護することが可能である．

3.4 秘密情報の作成

提案方式は，自己証明方式であるため，秘密鍵を何から容易に作成するかが重要になる．このとき，秘密鍵は作成者本人しか知りえない情報であることが求められる．容易に取得が可能な個人情報として，生体情報や使用機器の固有情報などが考えられる．前者には，掌紋や指紋などの生体情報があるが，現在まで認証機器が普及していないことや，使用環境等によるゆらぎ等による本人拒否や他人許容の問題がある．後者には，MAC(Message Authentication Code)アドレスやUSBメモリのシリアル番号(iSerial Number)などがあり，PCからの取得も容易である．

4．ハッシュを用いた自己証明方式の実装

前節に述べたハッシュを用いた自己証明方式を実現するため，表1のアルゴリズムに基づきドキュメントにハッシュを埋込んで検証を行うプログラムを作成した．

図3はPDFに対してハッシュを埋込んだ後に検証を行ったもので，文書のテキストに改ざんがない場合である．この場合には，埋め込んだハッシュ値と文書から計算したハッシュ値が一致するため，改ざんなしと表示される．

また，図4はWORDに対してハッシュを埋め込んだ後テキストに改ざんを加え，検証を行った場合である．このように，文書のテキストに改ざんが行われるとカスタムプロパティに埋め込まれているハッシュ値とテキスト及びsaltから計算されるハッシュ値が一致しないため，改ざんありと表示され，文書が改ざんや編集されたことを検出することができる．

表1　自己証明方式のアルゴリズム

埋込み
1. ドキュメントのテキストを抽出
2. テキストのハッシュ値を算出
3. MACアドレスを取得※
4. 秘密・公開鍵を作成してハッシュ値を暗号化※
5. カスタムプロパティに設定

検証
1. カスタムプロパティから値を取り出す
2. 公開鍵を使用して暗号化ハッシュを復号してハッシュ値を取得※
3. テキストのハッシュ値を算出
4. ハッシュの値を比較
　一致している場合：　改ざんなし
　一致していない場合：改ざんあり

※電子署名を行った場合のみ

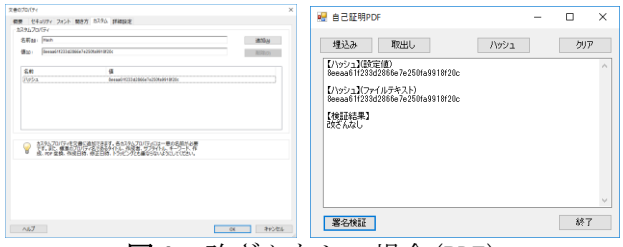

図3　改ざんなしの場合(PDF)

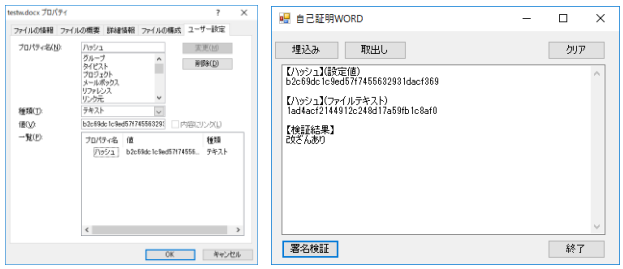

図4　改ざんありの場合(WORD)

5．まとめ

不正アクセスによるホームページやファイルの改ざんへの対策として，本論文では，ハッシュ値をファイルに埋め込むことで本文だけで自己の真正性を証明することが可能となる自己証明方式を提案した．

今後，本方式をドキュメントだけでなくインターネットで扱われる画像やプログラム等に応用し，リンクを含めたホームページの改ざんを簡易に検知する方式を提案したいと考えている．

参考文献

[1]朝日新聞:サイト改ざんや不正アクセス相次ぐ 大学,新聞社も被害,2016/11/21. http://www.asahi.com/articles/ASJCK5357JCKULZUOON.html
[2]YAHOO!ニュース:Avast 傘下の「CCleaner」にマルウェア混入,正規ルートで配信,2017/09/19.https://headlines.yahoo.co.jp/hl?a=20170919-00000030-zdn_ep-sci
[3]村中直樹:情報セキュリティアドミニストレータスーパー合格本,秀和システム,pp527,2008.
[4]増井敏克:おうちで学べるセキュリティのきほん，翔泳社,pp192,2015.
[5]上原孝之:情報セキュリティスペシャリスト,翔泳社,pp61,pp454,2016.