I-006

Sanitizable Signature の画像認証への応用
Application of Sanitizable Signature to Image Authentication

松本　侑 †　　伊藤　浩 †
Yuu Matsumoto　　Hiroshi Ito

1. はじめに

近年、電子署名はインターネット等の通信において真正性を保証するために広く利用されている。普通、電子署名が施されたデータに対しては変更を加えることはできない。この特性は電子署名で用いられる hash 関数によるものである。これに対して、Sanitizable Signature(以下 SS) [1] は、Chameleon Hash(以下 CH) 関数を使用した電子署名である。この署名方法は、hash 値を変えることなく正規の者は文書に変更を加えることができる。

本文では、SS を画像認証に応用する方法を提案する。この方法は jpeg の画像を入力とし、顔検出を行い変更可能と不可能の領域に分ける。変更可能領域には SS を施し、不可能領域には電子署名を施す。認証子とパラメータは jpeg の exif 情報に付与して、これを署名画像とする。CH 関数の入力に phash(perceptual hash) 関数 [2] を用いることにより、jpeg 再圧縮の耐性を持たせた。

2. Sanitizable Signature の原理

CH 関数は異なるパラメータから同一の hash 値を算出することができる hash 関数である。データ m、ランダム変数 (ρ, δ) を用いて、次式により hash 値を算出する。

$$c = \rho - (y^e g^\delta \bmod p) \bmod q \tag{1}$$

ここで、$e = H(m, \rho)$、$y = g^x$ である。また、H は一般的な hash 関数であり、本文では phash を用いる。g は素数巡回群の生成元、y は変更者の公開鍵、x は変更者の秘密鍵、p と q は素数である。

秘密鍵 x を所持する者は同一の hash 値になる (m', ρ', δ') の組を見つけることができる。最初に式 (1) と同様のパラメータを用いて c を算出し、m に変更を加えて m' とする。また、ランダムな値 $k' \in [1, q-1]$ を生成する。これらの値と式 (1) で用いたパラメータから以下の 3 式を計算する。

$$\rho' = c + (g^{k'} \bmod p) \bmod q \tag{2}$$
$$e' = H(m', \rho') \tag{3}$$
$$\delta' = k' - e'x \bmod q \tag{4}$$

この 3 式を式 (1) に適用すると以下のようになる。

$$\begin{aligned} c' &= \rho' - (y^{e'} g^{\delta'} \bmod p) \bmod q \\ &= c + (g^{k'} \bmod p) - (g^{xe'} g^{\delta'} \bmod p) \bmod q \\ &= c \end{aligned} \tag{5}$$

式 (5) より、c' は元のデータ m と同じ hash 値となる。このように hash 値が変化しないため、検証者は変更された文書に対しても通常の電子署名と同じように検証を行うことができる。

†日本大学, Nihon University

3. 提案方法

3.1 署名

図 1 は署名処理のブロック図である。署名を施す画像 (a.jpg) を入力とする。入力画像に対して復号を行ってから、顔検出を実行する。顔検出にはカスケード分類器を用いる。顔の矩形領域の位置情報を r とする。r は左上の点の x 座標、y 座標と領域の幅、高さである。顔部分を変更可能、それ以外の部分を変更不可能とする。

変更不可能領域に対しては phash 関数を使用して hash 値 (以下 phash 値) を算出する。この phash 値を鍵生成によって生成された署名者の秘密鍵 K_s^s を用いて暗号化を行う。生成された署名を S_i とする。変更可能領域に対しても phash 値を算出する。次に、この phash 値、ρ、δ さらに変更者の公開鍵 K_p^c を用いて CH 関数により hash 値 (以下 CH 値) を算出する。この CH 値も同様に秘密鍵 K_s^s を用いて暗号化を行う。生成された署名を S_m とする。生成された S_i、S_m、r、ρ、δ を入力画像の exif 情報に付与し、出力画像 (a'.jpg) とする。

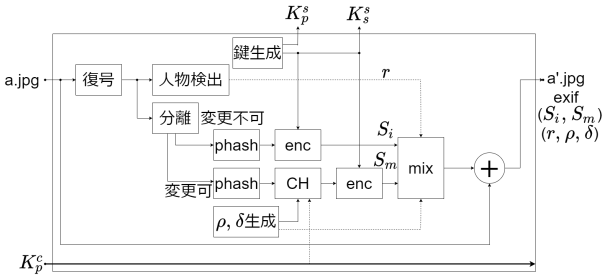

図 1: Block diagram of signing

3.2 検証

図 2 は検証処理のブロック図である。署名処理された画像 (a'.jpg)、署名者の公開鍵 K_p^s、変更者の公開鍵 K_p^c を入力とする。入力画像に対して復号を行ってから、r を用いて変更可能領域と不可能領域に分離する。

変更可能領域に対しては phash 値を算出する。算出した phash 値、公開鍵 K_p^c、ρ、δ を用いて CH 値を算出する。また、公開鍵 K_p^s を用いて署名 S_m を復号する。算出した CH 値と復号した CH 値を比較して、改ざん検知を行う。変更不可能領域に対しても phash 値を算出する。また、公開鍵 K_p^s を用いて署名 S_i を復号する。算出した phash 値と復号した phash 値を比較して、改ざん検知を行う。

3.3 変更

図 3 は変更処理のブロック図である。署名された画像 (a'.jpg) と変更者の秘密鍵 K_s^c を入力とする。入力画像に対して復号を行ってから、r を用いて変更可能領域と

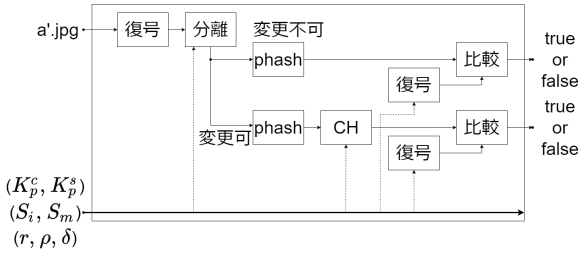

図 2: Block diagram of verification

不可能領域に分離する。変更可能領域は、変更前のデータ、公開鍵 K_p^c、ρ、δ を用いて CH 値を算出する。また、変更を加えたデータに対して phash 値を算出する。CH 値、phash 値、秘密鍵 K_s^c を入力として、式 (2)〜(4) により ρ'、δ' を算出する。次に、変更後のデータと変更不可能領域のデータを合成し、このデータに jpeg 符号化を行い、出力画像 (a".jpg) とする。exif の ρ、δ を ρ'、δ' に書き換えて、出力画像に付与する。検証者はこの画像を変更前のものと同じように検証できる。

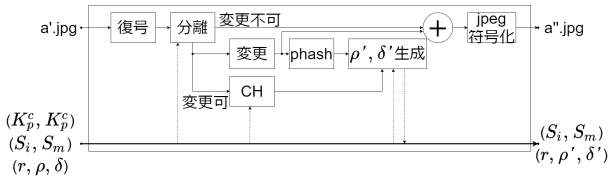

図 3: Block diagram of modification

4. 処理例

図 4 は提案方法に基づく画像の処理例である。素数 p と q は 3463 と 577 とした。また、公開鍵 K_p^c は 2305843009213693952 であり、対応する秘密鍵 K_s^c は 61 である。(a) は署名画像であり、変更可能領域には SS が施され、変更不可能領域には電子署名が施されている。CH 値は 102、phash 値は 561、ρ と δ は 687 と 48 である。(b) は顔検出器による変更可能領域を表示したものである。枠の内側が変更可能領域、線上と外側が不可能領域である。この領域は、変更プログラムが画面上で表示することが可能である。

(c) は変更者が変更を加えた画像であり、領域内の部分だけを黒塗りしている。変更後の CH 値は 102、変更不可能領域の phash 値は 561 であり、変更前と一致している。ρ' と δ' は 384 と 440 に書き換えられた。ここで、K_s^c を 50 とすると、ρ' と δ' は 384 と 302 になる。この値を用いて検証を行った場合、CH 値は 562 となり、102 と一致しない。これより、秘密鍵を持つ正規の者以外は変更を加えることができないことがわかる。

(d) は部外者が (c) の画像の黒塗り部分をさらに横に拡大した画像であり、変更可能領域の CH 値は 102 である。この領域は変更が加えられていないため CH 値は一致した。変更不可能領域の phash 値は 64 であり、横に拡大したため、もとの 561 と一致しない。これより、改ざんを検知することができた。

(a)Input image (b)Changeable range

(c)Modified image (d)Tampered image

図 4: Example

5. まとめ

署名者が電子署名を施した後でも正規の者は文書に変更を加えることができる SS を画像認証に応用する方法を提案した。実験の結果、署名、検証、変更の 3 段階全てで正しく動作することを確認した。また、不正な改ざん者が存在した場合でも改ざんを検知することができた。他の人物の画像を用いた場合でも検証できることを確認している。人物以外の画像も変更可能領域と不可能領域を設定すれば同様に動作する。複数人物が写った場合にも容易に対応することができる。

変更時の再圧縮により、変更不可能領域の phash 値が変化してしまうことがある。これは圧縮によってデータの損失が生じているためである。この問題を防ぐ最も単純な方法は再圧縮の画質劣化を少なくすることである。圧縮を行わないビットマップ等の画像形式を用いれば、解決されるが、ファイルサイズが大きくなる問題がある。また、逆に phash 関数は信号変化を見逃すことがある。ファイルサイズを増やさずに、これらに起因する誤認識を減らすことが今後の課題として挙げられる。

参考文献

[1] G. Ateniese, D. H. Chou, B. de Medeiros, and G. Tsudik “Sanitizable Signatures,” ESORICS, 2005.

[2] S. Wang, and X. Zhang “Recent development of perceptual image hashing,” J. Shanghai University (English Edition), vol.11, no.4, pp.323-331, 2007.