6X-04

USB型生体認証装置の安全性評価

橘　飛鳥†　梅原　直人†　岡本　剛†
神奈川工科大学†

1.はじめに

生体認証（バイオメトリクス）とは，指紋，虹彩，静脈など 1 人ひとりの固有の特徴を用いて機械が人間本人を確認する技術である．近年，生体認証装置の低価格化が進み，一般消費者や中小零細企業などでも容易に導入できるようになった．

生体認証装置は，生体情報を読み取る装置と認証を行う装置が一体となった装置と，それらが分離され USB ケーブルなどの通信ケーブルで接続する装置に分けられるが，後者は，通信ケーブルや PC から生体情報が窃取される可能性が考えられる．この脆弱性はバイオメトリクスのセキュリティ評価の国際規格 ISO/IEC 19792:2009 の第 8.3.11.1 項に該当する [1]．

そこで，本研究では，インターネットから容易に購入できる指紋認証装置 10 製品について，USB ケーブルで接続された認証装置から USB ケーブルに流れたデータを収集して，指紋画像の復元を試み，盗聴に対する安全性を評価した．

2.セキュリティ評価の方法

セキュリティ評価を行う手順は，評価環境の準備とデータ収集、データ解析からなる．評価環境の準備では，PC に認証を行うための認証アプリケーションをインストールする．認証アプリケーションは，個々の製品に標準で提供されているものを使う．次に，PC に USB アナライザをインストールする． データ収集では，生体情報を読み取る装置を PC に接続し，USB アナライザでデータの記録を開始し，認証装置で認証を行う．認証が完了したら， USB アナライザの記録を終了し、データを保存する．解析では，収集したデータから認証データと考えられるデータを抽出し，生体情報の画像の復元を試みる．

次節から，USB データの取得の方法，収集データをバイナリ化して，生体情報を画像化する方法を述べる．

2.1. USB 通信データの収集

認証装置がどのような内容を通信しているかを明らかにするため，認証装置が送信しているデータを収集する．データの収集には，USB アナライザを利用した．USB アナライザとは，USB デバイスの通信をログとして記録するためのソフトウェアやハードウェアのことである．本研究では，Windows XP 用の Sniffer for USB と、Windows 7 に対応している Device Monitoring Studio を用いた．

2.2. バイナリファイル化

前節で述べた USB アナライザの出力データはいずれもテキスト形式のため，データが画像であることを想定して，バイナリデータに変換する．

2.3. グレースケール画像化

本研究では，バイナリ化したデータをビットマップ方式の 256 階調グレースケールの画像化を行った．256 階調グレースケール画像は，カラー画像と同様に，鮮明な指紋画像が得られる．16 階調グレースケール画像は，256 階調に比べ，指紋のような鮮明な画像を取得できないと考えられるので，16 階調のグレースケール画像化は行わないことにした．指紋画像は，カラーでなくても，認証できると考えられるので，カラーの画像化は行わないことにした．

3.セキュリティ評価の結果

本研究では，インターネットから容易に購入できる指紋認証装置 10 製品について，セキュリティ評価を行った．なお，本稿では，安全性に関わる脆弱性情報を取り扱うため，セキュリティ評価を行う認証装置名は明らかにしないこととする．

3.1. 装置 A の結果

装置 A のデータを収集した結果，23 個からなる 2.9 メガバイトのデータを収集できた．このデータをすべてバイナリ化したデータのうち，ファイルサイズが 65 キロバイト以上の大きなバイナリデータを画像化した．その結果，指紋画像の復元に成功した．

3.2. 装置 B の結果

装置 B のデータを収集した結果，231 個からなる 3.6 メガバイトのデータを収集できた．このデータをすべてバイナリ化した．その結果，指紋画像の断片と考えられる 256×16 ピクセルの 18 枚の画像が得られた．これらの画像をつなぎ合わせた結果，指紋画像の復元に成功した．

Security evaluation for USB-based biometrics systems
†Asuka Tachibana, Naoto Umehara, Takeshi Okamoto, Kanagawa Institute of Technology

3.3. 装置 C の結果

装置 C のデータを収集した結果，109 個からなる 3.2 メガバイトのデータを収集できた．通信データに含まれる複数の 82 キロバイトのバイナリデータをすべてバイナリ化した．それらを結合して画像化した結果，指紋画像が得られた．ただし，ここで得られた指紋画像は，横方向に 75%圧縮されて，残り右側 4 分の 1 に，別の指紋画像が含まれていた．そのため，左 75%を横方向に伸張することにより、指紋画像を復元できた．

3.4. 装置 D の結果

装置 D は，装置 C と同様の結果が得られた．なお，装置 D と装置 C のメーカーは異なる．

3.5. 装置 E の結果

装置 E は，336 個からなる 599.9 キロバイトのデータを収集できた．このデータをすべてバイナリ化した．バイナリデータを画像化した結果，指紋画像は得られなかった．通信データを表示し，バイナリデータの規則性など調べたところ，送信時と受信時で同じ 16 進数列のパターン（0x43 0x69 0x61 0x6F 0x00）が現れる．その後に，送信データの 6 バイト目が，0x10 から始まり，0x10 ずつ増加し，7 から 8 バイト目には 0x01 0x30 のパターンが含まれる．受信データの 6 バイト目が，0x17 から始まり，0x10 ずつ増加し，7 から 8 バイト目には 0xFB 0x24 のパターンが含まれたが，これらが意味する内容は明らかにできなかった．

3.6. 装置 F の結果

装置 F は，認証アプリケーションがデバイスの内部にあり，PC に認証アプリケーションをインストールする必要がなく，生体情報を読み取る装置と生体情報の認証を行う装置が一体になった装置であった．したがって，通信中のデータから生体情報の漏えいの脆弱性はないと考えられる．

3.7. 装置 G の結果

装置 G は，6967 個からなる 103.1 メガバイトのデータを収集できた．このデータをすべてバイナリ化した．バイナリデータを画像化した結果，指紋画像は得られなかった．通信ログを表示し，バイナリデータの規則性などを調べたところ，64 バイトごとに 2 バイトのシーケンス番号のようなものがあったが，これが意味する内容は明らかにできなかった．

3.8. 装置 H の結果

装置 H のデータを収集した結果，128 個からなる 2.56 メガバイトのデータを収集できた．このデータをすべてバイナリ化した．ファイルサイズが 64 キロバイト以上の大きなバイナリデータを画像化した結果，指紋画像の復元に成功した．

3.9. 装置 I の結果

装置 I のデータを収集した結果，719 個からなる 14 メガバイトのデータを収集できた．このデータをすべてバイナリ化した．その結果，指紋の上半分と下半分の画像と考えられる画像を復元できた．これらを結合すると，指紋画像を復元できた．

3.10. 装置 J の結果

装置 J のデータを収集した結果，17,088 個からなる 12.8 メガバイトのデータを収集できた．すべてのデータが 1 キロバイト未満であり，バイナリ化したデータを画像化したが，指紋画像を復元できなかった．バイナリデータを分析した結果，カラー画像のフォーマットと類似したパターンが見られることから，カラー画像としての画像化を行う必要が考えられる．

4.装置の脆弱性と価格および発売時期の関係

本研究で明らかになった認証装置の脆弱性の有無と，それらの価格と発売時期の関係をに示す．この図から，価格と脆弱性に相関がないが，発売時期については，発売時期が新しくなると，脆弱性がない傾向がある．

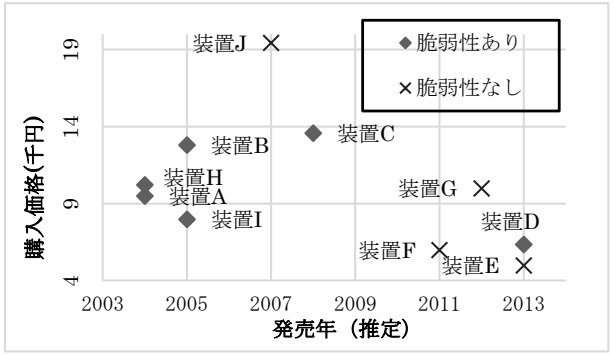

図 1 指紋認証装置一覧の散布図

5.おわりに

インターネットで購入できる指紋認証装置 10 製品について，セキュリティ評価を行った結果，6 製品に生体情報を窃取できる脆弱性を発見した．この脆弱性により，生体情報を窃取し，人工指紋 [2]などでなりすましなどの被害が考えられる．

参考文献

1) ISO/IEC 19792:2009, Security Evaluation of Biometrics, ISO/IEC Standard 19792, 2009.
2) GigaZiNE：iPhone 5s の指紋認証「TouchID」を突破する方法が判明, 2013, http://gigazine.net/news/20130924-how-to-hack-iphone-5s-touchid/