マルチメディア通信と分散処理 111－43
コンピュータセキュリティ 20－43
（２００３．２．２８）

ハッシュ関数を用いた生体情報による認証の実験

中村智博　　吉浦紀晃　　小野里好邦

近年、インターネットの普及による電子商取引(EC)などの増加から正確な個人認証が要求されている。現状では、個人認証を行う際、パスワードによる認証が主流だが、今後はネットワークを介した指紋認証や虹彩(アイリス)認証などの生体計測機器による認証が増加すると考えられる。しかしながら既存の生体計測機器による認証は独自の手法によりセキュリティの強化に励んでいるが、生体情報は暗号化されて保存されているが、その場合複合化され、生体情報が盗まれる可能性があるため、安全とは言い切れない。

そこで本論文では生体情報における認証時にハッシュ関数を用いて生体情報そのものを扱わない認証方法提案し、実験を行う。

Experiment of Biometric Recognition by Hash Function

Tomohiro Nakamura　　Noriaki Yoshiura　　Yoshikuni Onozato

In recent years,exact individual recognition is demanded from the increase in the electronic commerce by the spread of the internet etc. In the present condition, recognition with a password is main stream ,but biometric recognition through the internet increase from now on. Although biometric data is encrypted and saved in the existent recognition system, it may be decrypted, and then the biometric data itself may be stolen. This paper experiments of biometric recognition by proposing the method of treating the biometric data by hash function.

1. はじめに

近年、インターネットの普及により様々な情報がネットワークを介してやり取りされるようになっている。それに伴い電子商取引(EC)などの利用の増加により、正確な個人認証が要求される。現状では、パスワードによる認証が主流だが、今後はネットワークを経由したICカードによる認証や、生体計測機器による認証が増加すると考えられる。

しかしながら、生体計測機器から得られる情報は、個人そのものをあらわしており、悪質な第三者による盗難から守られなければならない。既存の生体計測機器による認証に利用される個人の生体情報の安全は、暗号化によって保持されている。しかし、復号化される危険性があるため、安全であると言い切れない。そこで、ユーザーの生体情報を安全に保護する条件の一つとして生体情報そのものが盗まれないということがあげられる。本研究では生体情報そのものを扱わないようにするため、一方向のハッシュ関数の関数[1]の利用を考えた。一方向のハッシュ関数は元のデータを推測されない一定長のメッセージダイジェストを生成する関数である。しかしながら、既存の指紋[2]やアイリス[3,4]認証などで用いられている鍵による暗号化をハッシュ関数を用いたものにそのまま置き換えてしまうと問題が生じてしまう。その問題というのは、生体計測機器による認証時に起こる登録時のデータと、認証時のデータに微小な誤差が生じてしまうことである。

〒376-8515 群馬県桐生市天神町　1-5-1 群馬大学大学院情報工学専攻.Dept. of Computer Science , Gunma University, Tenjin 1-5-1, Kiryu City ,Gunma,376-8515 Japan

〒376-8515 群馬県桐生市天神町　1-5-1 群馬大学総合情報処理センター. Computer Center, Gunma University,Tenjin 1-5-1 Kiryu City ,Gunma ,376-8515 Japan

〒376-8515 群馬県桐生市天神町　1-5-1 群馬大学大学 工学部　情報工学科. Computer Center, Gunma University,Tenjin 1-5-1 Kiryu City ,Gunma ,376-8515 Japan

既存の生体計測機器による認証は、登録時のデータと誤差の生じた認証時のデータとの差分を取り、それが許容範囲内であれば認証を成立しているため、誤差が生じても認証に影響を及ぼさない。しかし、ハッシュ関数は初期値敏感性(*1)を持つ関数のため、登録データと認証時のデータに微小な差が生じた場合、認証時に生成されたハッシュ値は、登録されているハッシュ値と全く異なってしまう。そのために、ハッシュ関数を用いた認証を行うことが非常に困難となってしまう。

以上から本論文では、その解決策を提案し、ハッシュ関数を用いた認証の実験を行う。

2 研究背景

本章では、ハッシュ関数、および生体計測機器による認証について記述する。

2.1 ハッシュ関数

ハッシュ関数とは、文字列や、数字の列を原文と全く関連性のない一定長のメッセージダイジェストを生成する関数である。ハッシュ関数の利用方法としては、通信回線を通じてデータを送受信する際に、経路の両端でデータのハッシュ値を求めて比較することにより、データが通信の途中で改竄されていないかを調べることができる。一方向関数を含むため、ハッシュ値から原文を再現することはできず、同じハッシュ値を持つ異なるデータを生成することは極めて困難である。このような特性から、認証やデジタル署名などに応用される。以下に代表的なハッシュ関数の例をあげる。

2.1.1 MD5:Message Digest 5

原文を固定長(128bit)のハッシュ値を生成する。計算方法には初期値敏感性の一方向関数を含むためハッシュ値は擬似的な乱数のような値をとり、これをもとにハッシュ値を生成する別のメッセージを作成することも非常に困難である。Ron Rivest 氏らによって開発され、RFC1321 として IETF で標準化されている。

2.1.2 SHA-1:Secure Hashing Algorithm 1

2 の 64 乗 bit 以下の原文から 160bit のハッシュ値を生成する。ハッシュ値は MD5 と同様に、擬似的な乱数のような値をとる。1995 年に米国標準技術局(NIST)によってアメリカ政府の標準ハッシュ関数として採用された。またインターネット上で安全に通信を行うための IPsec などに応用されている。

2.2 認証

本研究では指紋認証で用いられている指紋データを参考にしているため、ここでは指紋認証について記述する。

図 1 のように指紋とは隆線によって構成されている。特に隆線の切れ目を端点、分岐している点を分岐点と呼び、その二つを合わせて特徴点と呼んでいる。通常の指紋の場合、特徴点は約 50 個ほど存在し、この特徴点から得られる位置と方向を基本情報として扱っている。しかしこの情報だけでは、偶然に一致する可能性があるために、さらにこの特徴をはっきりさせるような情報を付加することによって一致する確率を下げる技術が考えられている。指紋認証とはこのような指紋の特徴や技術を利用して、本人確認を行うシステムである。

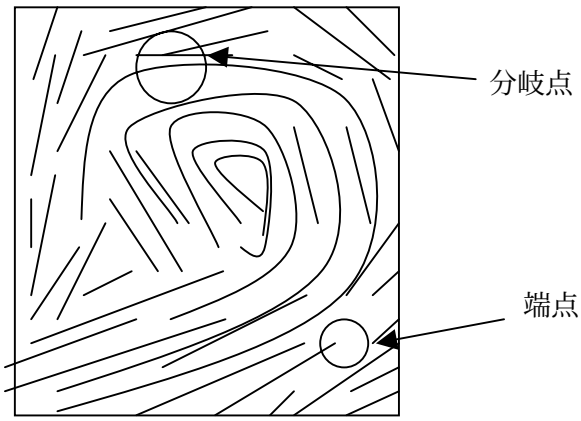

図 1：指紋

3 実験

(*1)入力(メッセージ)の微小な差によって出力が全く異なる
に原文を再現するのはほぼ不可能である。また同じハ

3.1 実験システム

実験の準備として以下のマシンを使用した。

	1	2	3	4	5	6
OS	Free BSD ver.4.6					
CPU	PentiumIV 1.5G	PentiumIV 2G				
HD	ATA 100 80G 7200 回転					
メモリ	512M	1G				

表:マシンスペック

また、既存の認証システムは安全面を考慮し、実際の生体情報や、生体情報の抽出方法を公開していないため、今回は既存の指紋認証で用いる指紋データを抽象化したものとして 4096bit のランダムな 1、0 の bit 列を生成するプログラムを C++言語を用いて構築した。

認証を実現するため、以下のシステムを構築した。

3.1.1 登録システム

前述したように、登録時と認証時のデータに誤差が生じても認証を行えるようにするため、指紋データに着目した。登録時と認証時で生じる微小な差とは、認証時に登録時とは異なった圧力で指紋データを読み取った場合、指の置く位置の微小なずれ、または、精神的な指の発汗などによって発生すると考えられる。本研究では、4096bit の中に誤差の発生する部分を決定し、その誤差の起こりうる全てのハッシュ値を登録しておくということを問題点の解決策として用いた。この実験では、4096bit の 1bit 目から 409bit 目、つまり先頭の 10%を誤差の出やすい領域とし、その中で生じる最大の誤差の bit 数を 4bit と設定する。すると以下の計算によって登録すべきハッシュ値の総数が算出される。

$$ {}_{409}C_4 + {}_{409}C_3 + {}_{409}C_2 + {}_{409}C_1 = 1160330955 $$

${}_{409}C_4$ は、誤差の生じやすい領域から 4bit の誤差が生じたときの総数であり、誤差が 3bit、2bit、1bit と全ての値をとることでハッシュ値の総数を導き出している。したがって、一人の生体情報を登録する際には約 12 億個のハッシュ値を登録することとなる。また、指紋データを読み取る際に、4096bit の生体情報に、ユーザー名(実験では生体データのファイル名とした)を付加した。より本格的に実装する際にはパスワードも付加する。このことから、もし登録データが盗まれても、ハッシュ関数を利用しているため、指紋データ、名前、パスワードは復元できない。このようにして、一人の指紋データを 6 台のマシンに分散させて登録するシステムを C++言語を用いて作成した。分散のさせ方としては、ずらす 409bit を 6 つのパートに分けて行っている。また、登録データの保存方法は、登録時に生成されるハッシュ値の先頭の 2bit についてソートし、00 から ff までの 256 個のファイルに分け、6 人分のハッシュ値を生成した。(図 2)

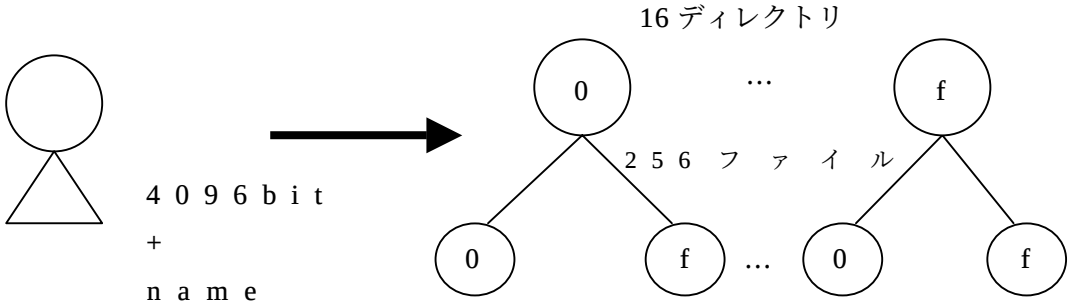

図 2：ハッシュ値の登録 DB

今回の実験では、ハッシュ値は通常一つ 128bit だが、今回の実装では 33byte になってしまった。したがって、12 億個のハッシュ値を生成すると 40Gbyte を要するため、マシンのスペック上登録人数を 6 人とした。

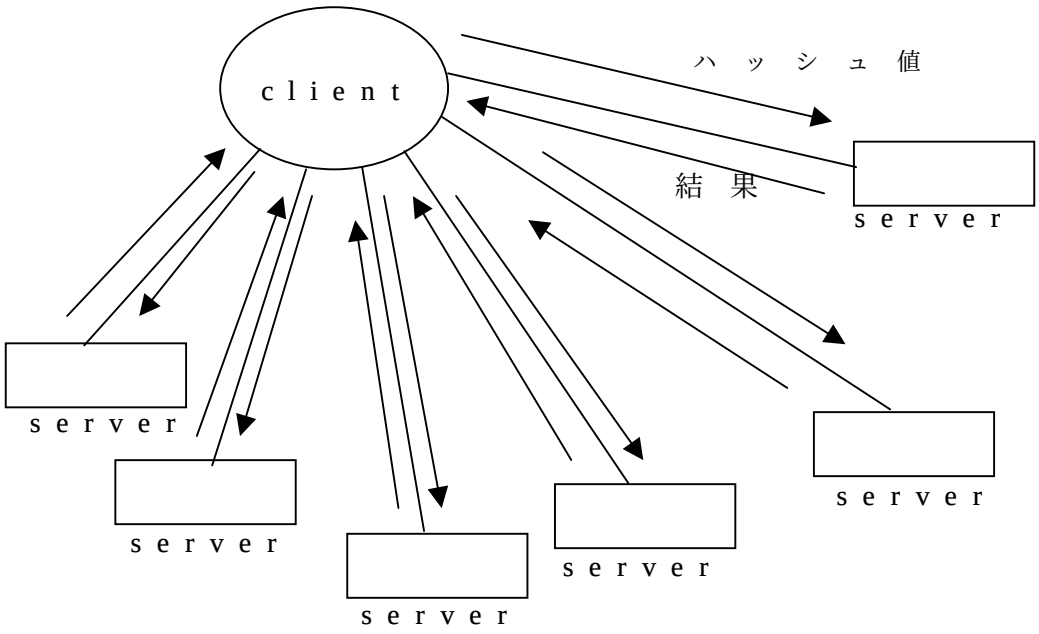

図３：サーバクライアントモデル

3.1.2 ネットワークを介した認証システム

ハッシュ値が保管されている 6 台の各マシンに、そのマシン上にあるデータを検索するサーバを実装した。また、それぞれのマシンに指紋データ(1,0 の 4096bit)と名前(ファイル名)を入力することによってハッシュ値を生成し、サーバに生成したハッシュ値を送信するクライアントを実装した(図 3)。この実験は、サーバとクライアントを実装することによって、ネットワークを介して認証を行うシステムである。

3.2 実験 1

登録システムを用いて一人の生体データを登録した。

3.2.1 結果と考察

マシン 1 では 17 時間、マシン 2 から 6 では、12 時間かかった。

マシン 1 と、他のマシンの登録時間の差はマシンのスペックによるものと考えられる。また、分散のさせ方としては、ずらす 409bit を 6 つのパートに分けて行っているので、マシンを倍にすれば、登録時間は半分になると考えられ、マシンのスペックを考慮すれば、登録時間の平均化も行える。しかしながら、現状は 409bit を 6 つのパートに分ける作業を異なったプログラムによって手動的に行われているが、今後は自動的に登録データの分散を行えるようにする。

3.3 実験 2

この実験では、生成した一人につき約 12 億個のハッシュ値が、他人の認証の際に誤って認証が成立してしまうかどうかを確かめる。検索方法は以下のとおりである。

1. MD5 によって、ハッシュ値を生成する。
2. ハッシュ値の先頭の 2 文字をチェックする。
3. 登録データに同一のデータがないかを検索する。
4. 1 で生成されたハッシュ値を再度 MD5 によってハッシュ値を生成する。
5. 以上をマシン 1 から 6 で、1 万回繰り返す。

今回の実験で 1 万回生成されたハッシュ値はそれぞれ異なるものであるが、検索開始時には同一のハッシュ値を用いているため各マシンで生成されているハッシュ値は同一のものとなる。

3.3.1 結果と考察

実験の結果、誤った認証は起こらなかった。
また、登録人数 6 人のデータの検索にかかった時間は、マシン 1 で 45 時間、マシン 2 から 6 では 35 時間となった。
今回の実験で誤って認証することはなかったが、今後はどのくらいの検索回数や、登録データを増やせば、誤った認証が行われるのか確かめる。

3.4 実験 3

今回の実験では一人につき大量なハッシュ値を生成しているため、それらのハッシュ値に重複がないかを shell スクリプトを用いて比較を行った。
具体的には以下のようにした。

Step1:sort 001 002 003 004 005 006 >00all
Step2:wc 00all
Step3:uniq 00all>00all2
Step4:wc 00all

ここでいう 001 というものは、マシン 1 の中に登録してあるハッシュ値の先頭が 00 の 6 人分のファイルを結合したものである。そして step2 step4 に差がなければ、全マシンに登録されている 00 で始まるハッシュ値の重複はないことがわかる。

3.4.1 実験 3-1

生体データが同一であり、生体データに付加する名前が異なっている 2 つのデータを登録し、比較した。

3.4.2 結果と考察

実験 3 の結果、重複したデータはなかった。また、3-1 も重複したデータは存在しなかった。それは以下のように考えられる。MD5 のハッシュ値は、32 文字の 16 進数で構成されており、異なるハッシュ値の総数は 16^{32} となる。したがって、ハッシュ値の総数は 72(12*6)億をはるかに上回るため、重複しなかったと考えられる。また、認証においては生体データの他に、名前や、パスワードなどを付加させることが一般的であるため、その付加情報によって個々のデータを分別しておけば、より高速で正確な認証が実現できると考えられる。

また、実験 3-1 の結果から、たとえ指紋データが近似しているものであっても、異なった名前を付加させただけで、異なった 12 億個のデータが生成できたことから、ハッシュ関数の信頼性も確認できた。

3.5 実験 4

認証システムを用いて、登録人数が 1 人から 6 人について認証時間にどのくらいの差があるかを計測した.

3.5.1 結果と考察

結果は下表のとおりである。
今回のシステムは、認証を行う際に、入力された生体データに名前を付加させ、MD5 によってハッシュ値を生成し、サーバ上のデータベースのデータと照合を行っている。表の結果から、登録人数が一人増すにつれて 2 秒から 3 秒かかっている。ネットワーク上を

登録人数	計測時間(sec)
1	3
2	5
3	7
4	10
5	13
6	15

表：認証計測時間

流れるデータは一つのハッシュ値分だけなので、通信速度にはほとんど影響がない。このことから、マシンをさらに増やし、分散させれば、各マシンの持つデータは減少するため、認証時間は短縮できると考えられる。さらに、ファイルの分割数の階層を増やせば、一つのファイルの容量は小さくなり認証時間が高速化する。また、この結果は認証の成否にかかわっていない。というのは、照合を行うプログラムは、はじめに受け取ったハッシュ値の先頭の 2 文字をチェックして、データベース上のどのファイルの中にあるかを決定し、照合を始める。しかし、照合の途中で認証が成立しても、そのファイルの最後まで照合を行っているためである。このことによって、認証成立時に終了すればさらに高速化できる。

4 まとめ

本論文では、暗号化を用いて保存しておくという方式に代わって、ハッシュ関数を用いた認証システムという、これまであまり報告されていないシステムを提案し、実験を行った。また、多人数の登録システムの可能性について実験し、考察した。また、今回は、既存の生体認証の実際のシステムや、生体情報がわからないために、生体計測機器から取得できる生体データをランダムな 1,0 の bit 列によって抽象化したデータを用いたため、誤差の許容範囲や、その中で生じる最大の誤差 bit 数の設定は推測、およびマシンのスペックに依存するものとなった。

今後の課題として、実際の生体情報を利用して生体情報のパターンからどの部分に誤差が顕著に表れるかがわかれば、既存のシステムのような認証を実現できると考えている。

また今回の認証システムは認証を行う際、接続するマシンが一台ずつなため、同時に接続できるようにする。また、今回用いたハッシュ関数である MD5 でなく、SHA-1 などの生成されるハッシュ値の bit 数が多ければより多人数に適したシステムを実現することが可能である。しかしながら、MD5 によって生成されるハッシュ値の総数は 2^{128} (16^{32})であり、一人に対する登録データを 12 億個とり、地球の総人口が登録してもその数には及ばないのである。このことから、認証に適したハッシュ関数というのも見つけてみたいと考えている。さらには、今回の実験結果において分散による効果が顕著に現れた。その結果と、認証の際に利用する送信データの小ささを利用し、広域分散型の認証システムも実装したいと考えている。

文献

[1] 渡辺治,”一方向関数のお話”,情報処理学会誌,Vol.32,No.6,pp.704-714

[2]TriCaohuu,LeThuy,”FingerPrintRecognition :Tecnology and Application”

[3]John Daugman,PhD,OBE “High Confidence Recognition of Persons by Their Iris Patterns”

[4] John Daugman,PhD,OBE “Iris Recognition Technology”