(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第6352381号
(P6352381)

(45)発行日　平成30年7月4日(2018.7.4)　(24)登録日　平成30年6月15日(2018.6.15)

(51)Int.Cl.　F I
H04L 9/32 (2006.01)　H04L 9/00 675C
G06F 21/31 (2013.01)　G06F 21/31

請求項の数 11　(全 19 頁)

(21)出願番号　特願2016-500901(P2016-500901)
(86)(22)出願日　平成26年3月7日(2014.3.7)
(65)公表番号　特表2016-512931(P2016-512931A)
(43)公表日　平成28年5月9日(2016.5.9)
(86)国際出願番号　PCT/US2014/022075
(87)国際公開番号　WO2014/150064
(87)国際公開日　平成26年9月25日(2014.9.25)
審査請求日　平成27年12月11日(2015.12.11)
審判番号　不服2016-16167(P2016-16167/J1)
審判請求日　平成28年10月28日(2016.10.28)
(31)優先権主張番号　13/844,619
(32)優先日　平成25年3月15日(2013.3.15)
(33)優先権主張国　米国(US)

早期審査対象出願

(73)特許権者　502208397
グーグル　エルエルシー
アメリカ合衆国　カリフォルニア州　９４０４３　マウンテン　ビュー　アンフィシアター　パークウェイ　１６００
(74)代理人　110001195
特許業務法人深見特許事務所
(72)発明者　バークマン，オメル
イスラエル、６９４００　テル・アビブ、ケヒラット・ベネツィア・ストリート、２
(72)発明者　ヨン，マルセル・エム・エム
アメリカ合衆国、１００１１　ニュー・ヨーク州、ニュー・ヨーク、ウェスト・トゥウェンティフォース・ストリート、２００

最終頁に続く

(54)【発明の名称】永続的認証のための、プライバシーを保護する知識／因子所有検査

(57)【特許請求の範囲】

【請求項１】

プロセッサを含む、装置であって、プロセッサは、

複数の質問への複数の応答から複数のハッシュを生成し、

認証ハッシュを、ノイズのある補間アルゴリズムを使用することで、複数のハッシュを対象とする多項式補間および前記複数のハッシュの各々が入力として使用されるハッシュ関数の演算から生成し、多項式補間のための１つ以上のエラー点、および多項式補間のための１つ以上の正しい点のうちの一方の導入を通して、前記ノイズのある補間アルゴリズムのしきい値を調節し、前記しきい値は、前記複数の応答のうち必要とされる正解の割合であり、

認証ハッシュを用いて認証するように構成されており、

前記認証ハッシュを格納するように構成されたメモリをさらに含み、

プロセッサは、複数のハッシュのうちの１つを認証ハッシュとして選択することに基づいて、当該選択から認証ハッシュを生成するように構成されており、

プロセッサは、前記複数の応答のうちの少なくとも２つから、前記複数のハッシュの各々を生成するように構成されている、装置。

【請求項２】

秘密認証ハッシュを格納するように構成されたメモリをさらに含み、

プロセッサは、秘密認証ハッシュと認証ハッシュとの比較を通して、認証ハッシュを用いて認証するように構成されており、さらに、

認証ハッシュが秘密認証ハッシュと整合する場合には、秘密認証ハッシュを認証に使用し、
認証ハッシュが秘密認証ハッシュと整合しない場合には、認証を拒否するように構成されている、請求項１に記載の装置。
【請求項３】
プロセッサは、認証ハッシュを形成するための複数の質問のうちの選択されたグループに関連付けられた複数の応答に対応する複数のハッシュの使用に基づいて、前記複数のハッシュのうちの一つ以上のハッシュの選択から認証ハッシュを生成するように構成されている、請求項１に記載の装置。
【請求項４】
プログラムであって、前記プログラムは、コンピュータのプロセッサに、
複数の質問への複数の応答から複数のハッシュを生成するステップと、
認証ハッシュを、ノイズのある補間アルゴリズムを使用して、複数のハッシュを対象とする多項式補間および前記複数のハッシュの各々が入力として使用されるハッシュ関数の演算から生成して、多項式補間のための１つ以上のエラー点、および多項式補間のための１つ以上の正しい点のうちの一方の導入を通して、前記ノイズのある補間アルゴリズムのしきい値を調節するステップとを実行させ、前記しきい値は、前記複数の応答のうち必要とされる正解の割合であり、
前記プログラムは前記プロセッサに、
認証ハッシュを用いて認証するステップをさらに実行させ、
認証ハッシュを生成することは、複数のハッシュのうちの１つを認証ハッシュとして選択することに基づいて、当該選択から認証ハッシュを生成することを含み、
複数のハッシュを生成することは、前記複数の応答のうちの少なくとも２つから、前記複数のハッシュの各々を生成することを含む、プログラム。
【請求項５】
複数のハッシュから認証ハッシュを生成することは、前記複数のハッシュを対象とする多項式補間を行なうことを含む、請求項４に記載のプログラム。
【請求項６】
認証ハッシュを用いて認証することは、
秘密認証ハッシュを認証ハッシュと比較することと、
認証ハッシュが秘密認証ハッシュと整合する場合には、秘密認証ハッシュを認証に使用することと、
認証ハッシュが秘密認証ハッシュと整合しない場合には、認証を拒否することとを含む、請求項４または５に記載のプログラム。
【請求項７】
複数のハッシュの各々は、複数の応答のうちの少なくとも２つから生成され、
複数のハッシュから認証ハッシュを生成することは、複数の質問のうちの選択された質問からなるグループに基づく認証ハッシュとして、複数のハッシュのうちの１つを選択することを含む、請求項４に記載のプログラム。
【請求項８】
プロセッサを含む、サーバであって、プロセッサは、
複数の質問を送信し、
送信された複数の質問に応答した認証ハッシュが秘密認証ハッシュと整合する場合には、アクセスを許可し、
認証ハッシュが秘密認証ハッシュと整合しない場合には、アクセスを拒否するように構成されており、
認証ハッシュは、ノイズのある補間アルゴリズムを使用することで、複数のハッシュを対象とする多項式補間および前記複数のハッシュにの各々が入力として使用されるハッシュ関数の演算から生成されて、多項式補間のための１つ以上のエラー点、および多項式補間のための１つ以上の正しい点のうちの一方の導入を通して、ノイズのある補間アルゴリ

ズムのしきい値を調節し、前記しきい値は、前記複数の応答のうち必要とされる正解の割合であり、
前記認証ハッシュを格納するように構成されたメモリをさらに含み、
プロセッサは、複数のハッシュのうちの１つを認証ハッシュとして選択することに基づいて、選択から認証ハッシュを生成するように構成されており、
プロセッサは、複数の応答のうちの少なくとも２つから、複数のハッシュの各々を生成するように構成されている
む、サーバ。
【請求項９】
プロセッサは、送信された質問のうちの選択されたグループに基づいて、複数の秘密認証ハッシュから秘密認証ハッシュを選択するように構成されており、
複数の秘密認証ハッシュの各々は、複数の質問のうちの少なくとも２つに関連付けられている、請求項８に記載のサーバ。
【請求項１０】
プロセッサはさらに、ユーザに関連付けられた装置およびユーザに関連付けられたアカウントから確認を受信後、秘密認証ハッシュを受信して、秘密認証ハッシュをメモリに格納するように構成されている、請求項８に記載のサーバ。
【請求項１１】
複数の質問は、生体計測情報についての要求を含む、請求項８から１０のいずれか１項に記載のサーバ。
【発明の詳細な説明】
【技術分野】
【０００１】
背景
１．技術分野
例示的な実施形態の局面は、永続的認証のためのプライバシー保護検査に関し、より特定的には、プライベートな質問に対する回答がサーバ側で暴露されないように、認証ハッシュまたは他の一方向性の反転しにくい関数を生成し、生成された認証ハッシュに基づいて認証するための装置、方法、およびシステムに関する。
【背景技術】
【０００２】
２．関連技術
アクセスのための、およびアカウントの回復のためのユーザの認証プロセス中のさまざまな状況において、ユーザが自分の身元を証明しなければならない場合がある。認証、または認証のための代替的方法（たとえば、故障許容／回復）を容易にするために、ユーザは、因子（ファクター：factor）（たとえば、ユーザの生活および趣味に特有の質問への回答）を、アクセスを保持するサーバ（たとえば、アカウントプロバイダ）に登録する。回答を含む、ユーザによる登録は、プライベートなユーザ情報をサーバに暴露するおそれがある。悪意のある第三者による、サーバへの不正アクセスは、プライベートなユーザ情報をその悪意のある第三者に暴露するおそれがある。たとえば、その第三者（たとえば、サーバ組織のインサイダー、もしくはアウトサイダーまたはフィッシング攻撃者）は、同様の回答を必要とし得る他のまたは同じアカウントプロバイダで登録された回答を利用して、そのユーザになりすますおそれがある。
【０００３】
認証のために、ユーザが質問に回答すること（または、生体計測情報、システム外部に格納された所有された情報といった他のプライベートな因子を提供すること）を可能にしつつ、プライバシー上の理由により、ユーザからの情報を検証するサーバに個人情報を保持させないようにするという要望が存在する。
【発明の概要】
【課題を解決するための手段】

【０００４】
概要
本願の局面は、プロセッサを伴う装置であって、プロセッサは、複数の質問への複数の応答から複数のハッシュを生成し、認証ハッシュを、複数のハッシュの多項式補間と、複数の質問のうちの選択されたグループに基づいて認証ハッシュを形成するための、複数のハッシュのうちの１つ以上の選択との少なくとも一方から生成し、認証ハッシュを用いて認証するように構成されている、装置を含み得る。
【０００５】
本願の局面はさらに、プロセスを実行するための命令を格納する、コンピュータ読取可能記憶媒体を含む。命令は、複数の質問への複数の応答から複数のハッシュを生成することと、認証ハッシュを、複数のハッシュの多項式補間と、複数の質問のうちの選択されたグループに基づいて認証ハッシュを形成するための、複数のハッシュのうちの１つ以上の選択との少なくとも一方から生成することと、認証ハッシュを用いて認証することとを伴い得る。
【０００６】
本願の局面はさらに、プロセッサを含み得るサーバであって、プロセッサは、複数の質問を送信し、送信された複数の質問に応答した認証ハッシュが秘密認証ハッシュと整合する場合には、アクセスを許可し、認証ハッシュが秘密認証ハッシュと整合しない場合には、アクセスを拒否するように構成されており、認証ハッシュは、複数のハッシュの多項式補間と、複数の質問のうちの選択されたグループに基づいて認証ハッシュを形成するための、複数のハッシュのうちの１つ以上の選択との少なくとも一方から生成される。
【図面の簡単な説明】
【０００７】
【図１（ａ）】例示的な一実現化例に従った、装置についてのフロー図を示す図である。
【図１（ｂ）】例示的な一実現化例に従った、装置についてのフロー図を示す図である。
【図２（ａ）】例示的な一実現化例に従った、サーバについてのフロー図を示す図である。
【図２（ｂ）】例示的な一実現化例に従った、サーバについてのフロー図を示す図である。
【図３】いくつかの例示的な実現化例で使用するのに好適な例示的なコンピューティング装置を有する、例示的なコンピューティング環境を示す図である。
【図４】例示的な実現化例に従った例示的な処理環境を示す図である。
【発明を実施するための形態】
【０００８】
詳細な説明
ここに説明される主題は、例示的な実現化例によって教示される。明確にするために、および主題を不明瞭にしないようにするために、さまざまな詳細が省略されている。以下に示す例は、プライバシー保護を有するキャンペーンパフォーマンスの測定を実現するための構造および機能に向けられている。例示的な実現化例の局面は、たとえば、電子商取引、情報共有、プライバシー保護方法、符号化および暗号化手法、トランザクションシステム、個人情報共有、および、セキュアなコンピューティングに関し得る。しかしながら、例示的な実現化例はそれらに限定されておらず、この発明の概念の範囲から逸脱することなく他の分野に適用されてもよい。
【０００９】
ここに説明される例示的な実現化例は、サーバ（またはサーバの情報を有する誰か）がプライベートな回答を推定できるようにする情報をサーバ側で暴露しない、ユーザのプライベートな因子に基づいた認証に向けられている。例示的な実現化例では、ユーザは認証因子を用いて質問票に回答し、回答は一方向性のやり方で変換され、変換された回答はサーバ側に提供される。これは、ユーザのプライバシーを保護しつつ、サーバがサーバに情報を登録した元のユーザを認証できるようにする。

【００１０】
例示的な実現化例は、十分なエントロピー（たとえば文字列）を有する複数の因子が、ユーザ装置上で一方向性（たとえば暗号学的ハッシュ）関数の下でともに変換されるようにすること、および、変換された値を登録時にサーバに送信することに向けられている。認証セッションで、ユーザは再度回答を求められ、それらの回答は上述のものと同様のやり方で装置によって変換され、サーバに送信される。サーバは次に、一方向性で変換された回答を、登録された情報と比較する。以下の説明は、例示的な実現化例で使用される機構を概説する、より詳細なプログラム／プロトコルに向けられている。
【００１１】
例示的な実現化例についてのプロトコルエンティティは、ユーザ、ユーザ装置、およびサーバを含んでいてもよい。明確にするために、プロトコルパラメータはｎ、ｔ、ｒおよびｍとして表わされ、それらについて以下に説明する。
【００１２】
例示的なプロトコル環境では、以下に説明されるような、考慮すべきいくつかの局面がある。
【００１３】
プライベートな登録情報：ユーザは、個人情報のｎ個のラベル付き文字列を有する。これは、ユーザが知っていて思い出しそうな何か、または、ユーザが所持または所有する因子であり得る。例示的な実現化例では、初期登録は、たとえば、ユーザに関連付けられた装置（たとえば、ユーザの電話、および代替的電子メールなどのユーザに関連付けられたアカウント、または友人のアカウント）に送信された受領通知をユーザが受信し、応答した後で、有効になり得る。
【００１４】
プライベートでないラベリング：文字列のラベル、フォーマット、おそらくはヒント、および順序は、プライベートではない。
【００１５】
永続性：いかなる時も、ユーザは、少なくともｎ－ｔ個の文字列を知っている。すなわち、ユーザはｎ個の文字列を登録したかもしれないが、それらのすべてを常に思い出すとは仮定できず、それらのうちのｔ個を忘れるかもしれない。このため、ユーザは、文字列のうちのｎ－ｔ個についての何らかのしきい値を知っていることが必要とされる。なお、必要とされるレベルは、認証セッションごとにサーバによって調整可能である。
【００１６】
ユーザ装置：ユーザは、データのセキュアな入力、計算、データの消去、データの保存、およびデータの出力が可能な装置へのアクセスを有する。装置は、ユーザの制御下にある（たとえば、それはフィッシング不可である）。これは、ウェブに接続されていないスマートフォンまたはソフトウェア要素であり得る。
【００１７】
装置の部分的完全性：装置は正確に動作する（特に、要求された場合、データは永続的に消去される）が、失われる／盗まれるかもしれない。
【００１８】
サーバの完全性：サーバは正確に動作し、データを決して失わない。なぜなら、サーバはユーザを認証することに関与しているためである。また、サーバで長期間格納されたデータは、攻撃者らがユーザになりすますことを可能にするデータを含んでいない。
【００１９】
セットアップ：セットアップ中、装置およびサーバは、情報をセキュアに交換できる。
例示的なプロトコル環境はまた、以下のようないくつかの要件を含み得る。
【００２０】
プライバシー：プライベートな文字列のうちのｒ個について知っていると仮定すると、サーバ上、装置上の情報、またはサーバと装置との間で交換された情報は、残りのｎ－ｒ個の文字列のうちのいずれかを暴露するのに、または、残りのｎ－ｒ個の文字列のうちの

いずれかを最初よりも良好に推測するのに実質的に不十分であるはずである。

【００２１】

真正性：いかなる時も、ユーザは、ユーザが入力文字列のうちの少なくともｎ－ｔ個（ｎ－ｔはｒよりもはるかに大きい）を知っているということを（装置を用いて）サーバに証明することができる。この真正性動作は動作の好結果を決定し、サーバは、さまざまな認証セッションにおいて、必要とされるしきい値ｎ－ｔをおそらく動的に変化させてもよい。

【００２２】

セキュリティ：サーバ上、ユーザによって使用されていない装置上の情報、またはサーバと装置との間で交換された情報は、最初に登録した元のユーザでないかもしれないユーザを認証するために使用するのに実質的に不十分であるはずである。

【００２３】

例示的なプロトコル環境は、さまざまなプロトコルを採用してもよい。たとえば、さまざまな因子をセットアップするために、因子登録が行なわれてもよい。因子登録は、ランダム化、作表、回答および生成を伴っていてもよい。

【００２４】

因子登録のランダム化局面では、装置およびサーバは、乱数発生器または他の方法を用いて、ランダム性を共同で生成してもよい。例示的な一実現化例では、サーバは、長いランダム（秘密でない）ソルト（salt）Ｒｓを装置に提供する。ユーザは、長いランダム（秘密でない）ソルトＲｕを生成し、ＲｓおよびＲｕを装置に入力してもよい。装置は、長いランダム（秘密でない）ソルトＲｄを生成し、３つのランダムソルトすべてを単一のランダムソルトＲへと連結する（ソルトＲは、さらなるインタラクションにおいてサーバによって採用されるべき因子であってもよい）。

【００２５】

因子登録の作表局面では、ラベル提供が行なわれてもよい。サーバはユーザに、１組の文字列ラベルと、各文字列のそれぞれの可能なフォーマットと、ユーザによって採用されるべき１組の基準「ヒント」とを提供する。ラベルとは、ユーザがある文字列において所与のフォーマットで値を提供する、変数である。ユーザは、質問票を規定するために、提供された順序付けられた一組のラベルからｎ個のラベルを選択してもよい。例示的な実現化例では、質問票におけるいくつかの要素は必ずしも「知っている何か」というタイプのものではないかもしれず、他のタイプの情報（たとえば、生体計測、カスタマイズされた質問など）が同様に使用可能である。

【００２６】

因子登録の回答局面では、ユーザは、質問票の回答をｎ個の文字列として提供する。ユーザは、プロセスの一環として回答を繰り返すよう、システムによって訓練され得る（たとえば、ユーザは２度質問され、システムは回答のユーザ記憶を増加させるための手法を採用するなど）。回答は装置に移動され得る。

【００２７】

質問票は、所望の実現化例に依存して、秘密のままにしておいたり、または、オープンである（たとえば、サーバがそれに対する直接的な回答を知っている）他の方法と混合されることができる。たとえば、質問票は、（たとえば、唯一の方法であるというよりはむしろ、識別についてのクレームの機能を高めるように）他の認証方法と組合せて使用することができる。たとえば、この組合せは、他の方法の失敗時、他の方法を用いた何らかの最初の成功後、他の方法が使用される前、ユーザがすでに認証されているもののさらに高感度のアクセス／行動を要求する場合のみ、使用可能である。

【００２８】

因子登録の因子生成局面では、システムは、回答に基づいて、およびアルゴリズムを利用することによって、思い出すべき因子を生成する。アルゴリズムを初期化するために、装置にはｎ個のユーザ秘密ｕ＿１、…、ｕ＿ｎが与えられ、ｕ＿ｉ＝質問ｑ＿ｉおよび回答ａ＿ｉである。装置はｎ個の秘密ｓ＿１、…、ｓ＿ｎを生成し、それらは、ｑ＿ｉのハ

ッシングまたは一方向性関数である。ｓ＿ｉ＝ＨＡＳＨ（ａ＿ｉ，Ｒ）。装置は、ｉ＝１、ｎについての点（ｑ＿ｉ，ｓ＿ｉ）から、平面におけるすべての点を通過するｎ－１次の多項式Ｐを、補間によって有限体上に生成することができる。ｑ＿ｉおよびｓ＿ｉの各々は有限体において解釈され、たとえばハッシュは、サイズが２５６ビットの素数を法として、その素数によって規定された有限体における要素として解釈された、２５６ビットの文字列であってもよく、ハッシングを介して生成されたｑ＿１およびｓ＿ｉは、有限体にＸおよびＹ座標を有するデカルト平面に位置する点として見ることができる、ランダムに見える点にマッピングされるであろう。有限体、素数、および多項式補間は、当業者にとって基本的な概念である。秘密ｓは、０での多項式の値であり（すなわち、Ｐ（０）＝ｓ）、シリアルナンバーとともにサーバで登録することができる。加えて、点（１，Ｐ（１））、（２，Ｐ（２））、…（ｋ，Ｐ（ｋ））といった、多項式上の追加のｋ＝２ｔ個の点が、これらは補間で元々使用された点ではないと仮定して送信され、サーバで登録される。これは、将来の認証においてユーザが間違っていること、または、ｎ個の文字列から可能なｔを省略することを可能にするｉである。多項式Ｐは（ｎ個の点によって生成されたため）ｎ－１次を有しており、点（ｏ，Ｐ（０））である秘密と追加されたｋ個の点との登録は、ｋ＋１がｎよりも小さくなるべきであり、このｋ＋１個の点の知識はサーバに多項式の特性を与えない。たとえば、将来１５個の回答しか必要としないというしきい値を保有しつつ、ユーザが回答すべき２０個の因子について尋ねられる場合、秘密に加えて１０個の点がサーバに送信される。ユーザが将来認証される（例示的な一実現化例として以下に説明する）と、ユーザは因子を再度送信し、追加された１０個の点は多項式の表現に追加され、これらの点を含むノイズのある補間がユーザによって試みられ得る。攻撃者がなりすましを試みる場合、攻撃者の知識は常に１０個の点よりも少ないであろう。なぜなら、因子は、因子についてのユーザ知識および所有を表わすように注意深く選択されたためである。このため、サーバによって送信された点、およびなりすましを試みる者の知識は、多項式Ｐを復元するために利用可能な点を補間できないであろう。

【００２９】

別の例示的な実現化例では、ｓ自体ではなくＨＡＳＨ（ｓ）がローカルに維持される。所望の実現化例に依存して、他の情報が装置によって削除または維持されてもよく、もしくは、ｋ個の点がサーバで維持されてもよい。たとえば、他の情報を消去すると、ユーザは、認証時に情報を再度入力しなければならず、一方、情報を維持することは、装置の所有を証明する際に使用可能である。ＨＡＳＨは、任意の一方向性関数、暗号学的ハッシュアルゴリズム、もしくは、暗号文書でのべき乗剰余について知られているような、ある有限体または別の代数構造上の発生器を有するべき乗であってもよい。ｓではなくＨＡＳＨ（ｓ）をサーバで保持することは、サーバに侵入する攻撃者らがｓ自体を学習することを防止する。

【００３０】

例示的な一実現化例では、認証セッションが、以下に説明されるように採用されてもよい。認証セッションは、因子のさまざまな使用モードを含んでいてもよい。第１のモードでは、装置は利用可能であり、ユーザは装置へのアクセスを有しており、秘密ｓは削除されなかった。第１のモードで、装置は次に、ハッシュのシリアルナンバーをサーバに通知し、セキュアプロトコルを使用することによって秘密についての知識を証明する。

【００３１】

第２の使用モードでは、サーバ、ユーザ、および装置（または別の装置）が協力し合って、ハッシュのうちの１つを生成する。サーバはユーザに、ｎ個のラベル（質問）とそれらのフォーマットとを送信する。次に、サーバは装置に、ｎ個のラベルとソルトＲとを送信する。ユーザは装置に回答ａ＿ｉを入力する。サーバは、ｋ個の追加点（１，Ｐ（１））、…（ｋ，Ｐ（ｋ））も送信する。ノイズのある補間アルゴリズム（たとえば、バーレカンプ・ウェルチ（Berlekamp Welch）、グルスワミ・スーダン（Guruswami－Sudan）など）を使用して、装置は多項式を計算し、回答のしきい値が正しい場合（たとえば、上述の２０個中１５個の例でのような２／３、半分など）、ノイズのある補間アルゴリズムは

ｓを生成する。装置がＨＡＳＨ（ｓ）を有する場合、生成されたｓは、正しいかどうかチェックを受けることができ、また、ユーザに新しい回答を求めてもよい（たとえば、正しくない場合、初期化などについて）。結果として生じるｓはサーバに送信され、サーバはユーザを認証し、またはそれに代えて、ユーザの装置は、サーバに送信されるＨＡＳＨ（ｓ）に基づいてｓの所有を証明し、この目的を達成するために、当該技術分野で公知のゼロ知識プロトコルまたはチャレンジ応答プロトコルを利用することができる。

【００３２】

ノイズのある補間アルゴリズムについての点のうちの１つが（たとえば、サーバまたはローカルソフトウェアによって付与された）ランダマイザである場合には、結果として生じる因子はランダム化される（すなわち、ユーザの回答から独立している）。たとえば、まず、バーレカンプ・ウェルチのノイズのある補間アルゴリズムからの２／３という範囲が、サーバにいくつかの点を付与させることによって調整可能であると仮定する。より高いしきい値が所望される場合には、サーバは（多項式上にない）エラー点を付与することができる。したがって、所望のしきい値がたとえば１８個中１６個（１６／１８）の点であり、バーレカンプ・ウェルチのノイズのある補間アルゴリズムが採用されている場合、２４個中１６個（１６／２４）の点が正しくなるように、サーバまたは装置のいずれかによって６つのエラーが導入でき、それによりバーレカンプ・ウェルチのしきい値を満たす。別の例では、実現されたしきい値が、回答の半分のみが正しいことを必要とする場合には、サーバまたは装置によって「良好な多項式点」が導入できる。たとえば、１８個中１０個（１０／１８）（質問の半分以上が正しい）が十分であると考えられる場合、結果が１６／２４になるように６個の良好な点を導入でき、それは２／３というバーレカンプ・ウェルチのしきい値を満たす。必要とされるしきい値の調整は、認証セッションごとに変わり得る。

【００３３】

選択された情報は非常にプライベートなものであるため、ユーザは、情報のほとんどすべてを思い出せるはずである。複雑性は、有限体における多項式の評価の複雑性である。

【００３４】

文字列は非常にプライベートなものであり得、必要とされる場合にユーザがそれらのほとんどを思い出せることを確実にする秘密情報を伴い得る。例は、所望の実現化例に依存して、兄弟、子供、配偶者、両親、祖父母、友人の名前、自分および親類のアドレス、アカウント名および／または番号、雇い主などの名前を含む。文字列についての選択基準は、必要とされる場合にユーザが回答を再作成できるようなものであるべきである。データの量および可変性は、十分な文字列が攻撃者に決して知られず、そのため、サーバからの追加点を有していても、攻撃者は良好な補間点を生成できず、多項式が攻撃者にとって秘密のままである、といったものであるべきである。

【００３５】

例示的な実現化例では、いくつかのセキュリティレベルも導入可能である。たとえば、文字列のラベル、フォーマット化および順序は、それら自体が、基本的で思い出しやすいいくつかの文字列（たとえば、ユーザのパスワード）によって保護されてもよい。

【００３６】

アカウント回復および乗取り犯によって乗取られたアカウントの解除の目的のために、回復プロセスのために使用され、以下の特性を有する認証因子が採用されるべきである。

【００３７】

永続的である：ユーザにとって常に利用可能である。ユーザは、因子を含む物理的対象を失っても、または（たとえば乗取りによって）自分のアカウントを失っても、それを失うことができない（もしくは、それを再作成することができる）。

【００３８】

偽造不可である：アカウントまたは個人ユーザ情報へのアクセスが与えられた場合でも、実質的に推測できない。ランダム攻撃者らおよびユーザの仲間双方に対して、偽造不可であるべきである。

【００３９】
プライベートである：アカウントプロバイダまたは攻撃者に個人データを暴露しない。
可用性を有する：特殊用途装置を用いない汎用ソフトウェアシステムにおいて実現可能である。
【００４０】
因子を選択するための検討事項がいくつかある。たとえば、永続的因子が「ユーザが有する何か」である場合、ユーザはその因子を失うかもしれず、または、その因子は攻撃者の手に渡るかもしれない。永続的因子が「ユーザが知っている何か」である場合、その因子は、チェックするシステムにとってプライベートなものではないかもしれず、ユーザはその因子を忘れるかもしれない。永続的因子が「ユーザの状態である何か」である場合、その因子は人間の何らかの特徴認識（生体計測装置など）を必要とし、容易には利用できないかもしれず、また、個人情報をプロバイダに暴露するかもしれない。
【００４１】
例示的な実現化例では、永続的因子はユーザの知識（「知っている何か」）に基づいて利用されており、また、ユーザが所有する何かに基づいていてもよい。そのような要件は、既存の状況の多くでは、満たすことが困難かもしれない。したがって、例示的な実現化例は、ユーザが多くの基本的質問を確実に思い出せると仮定し、回答を暗号演算と関わらせて、ユーザの知識に基づいた解決策を伴い得る。
【００４２】
自分および他人の知識：例示的な実現化例は「ユーザの知識」に基づくものとして提示されているが、その知識は、受託者および他のソースからリアルタイムで取得可能であり、知識の蓄積は、ユーザの個人的な知識および受託者へのユーザアクセスを表わし得る。受託者は、ユーザについての知識の一部を表わし、必要とされる因子をユーザが生成することを助けることができる。
【００４３】
例示的な実現化例は、アカウントへのアクセスの緊急回復のための基本的プロセスを伴い得るが、プライバシーと真正性とのバランスを取って有用性を考慮に入れる一般的な認証方法としても実現可能である（たとえば、因子が必要とされる場合にユーザを訓練するユーザトレーニングおよびユーザインターフェイス）。
【００４４】
ユーザがインターネットアカウントプロバイダから有するアカウントは、ユーザが自分の電子メール、電子支払い、個人のコンテンツなどをアカウントに保持するにつれて、重要性が高まっている。これらのアカウントは主要な個人リソースであり、攻撃者らの影響を受けやすい。例示的な実現化例は、ユーザは常に利用可能であるが攻撃者は決して利用できない永続的認証因子をユーザが有する場合に、ユーザが、乗取り犯には不可能なやり方でアカウントを保有し、再クレームすることができるようなシステムおよび方法に向けられている。そのような因子を近似することは、回復プロセスを緩和し得る。
【００４５】
関連技術では、電子メールアカウントなどのアカウントが乗取られると、攻撃者はアカウントの状態を有し、ユーザである悪意のないアカウント保持者による回復がより困難になるようにアカウントを操作できる、ということは事実である。攻撃者はまた、アカウントに格納されたすべてのデータから学習することができる。例示的な実現化例はしたがって、アカウントへのアクセスを有することから推論できない機構を利用する。同様に、それらの機構は、アカウントが利用できない場合（たとえば、乗取られた場合）にそれらが失われない、といったものであるべきである。回復が次に、永続的因子の保持者によって支配される。
【００４６】
例示的な実現化例は、ユーザの知識、または、必要とされる場合にユーザが再作成できる知識の高エントロピーソースを採用する。この目的のために、兄弟、子供、配偶者、両親、祖父母、友人の名前、自分および親類のアドレス、アカウント名および／または番号

、雇い主の名前などの、非常にプライベートな大量のユーザ情報が利用される。この情報は、必要とされる場合にユーザが回答を再作成できるようなものであるべきであり、データの量および可変性は、十分なビットが攻撃者に決して知られないようなものである。同様に、生体計測読取りまたは銀行サーバなどの受託者へのアクセスといった他の因子も、攻撃者に知られずに、組合わされると仮定される。

【００４７】

別の例示的な実現化例では、因子は、生成のための、および因子をチェックするための入力、処理および出力を有するプロセスによって生成可能である。このプロセスは、各々役割を有する、ユーザー入力、システム入力、および暗号演算を伴い得る。

【００４８】

入力は、ユーザが尋ねられるＱ１、Ｑ２、Ｑ３などの質問と、Ａ１、Ａ２、Ａ３などの回答との組といった、知識の高エントロピーソースを伴い得る。回答Ａｉは、ユーザが思い出せるようなものであるべきである（質問は何度も尋ねられ、ユーザはそれにより、質問に回答するように訓練され得る）。そのような質問の選択は、生活の質問、（さまざまな分野における）趣味の質問、個人履歴の質問などを伴い得る。さらに、質問の数は、所望のエントロピーを生成できるほど十分に多いものであるべきである。所望の実現化例に依存して、ユーザが携帯装置または１枚の紙の上に保持する、または、ユーザにメールされてインターネットアカウントの外部で保持されるランダム値Ｒ１、Ｒ２など、および／または、ユーザのローカルシステムが保持する追加されたランダム値、および秘密Ｓも採用可能である。

【００４９】

処理は、要因生成を伴い得る。質問への入力である回答Ａ１、Ａ２、…Ａｎが（繰り返すことで）グループへと体系化され、たとえば、Ｇ１＝Ａ１、Ａ３、Ａ５、Ｇ２＝Ａ１、Ａ３、Ａ６、Ａ７となると仮定する。１つのグループは、ユーザが全部回答するるよう期待されている、１組の連結された回答を表わす。ｍ個のグループがあり、各グループが十分に高いエントロピーを有すると仮定する。所望の実現化例に依存して、ランダム値Ｒ１、…、Ｒｍおよび秘密Ｓも追加（連結）されてもよく、たとえばＳが各グループに、ＲｉがＧｉに追加され、そのためＧ１＝Ｓ、Ｒ１、Ａ１、Ａ３、Ａ５となる。

【００５０】

各Ｇｉは、暗号学的ハッシュ関数（たとえばＳｈａ１など）Ｈを用いてハッシュされる。たとえば、Ｈ１＝Ｈ（Ｈ（Ｈ（Ｈ（Ｈ（Ｒ），Ｓ１）Ａ１）Ａ３）Ａ５）である。演算を遅くするための追加のハッシングも行なわれてもよい。Ｈ＿ｉは指標（インディケータ）と呼ばれる。

【００５１】

各グループは、それ自体の指標Ｈｉ：Ｈ１、Ｈ２、…、Ｈｍを有する。ランダマイザと呼ばれるランダム値Ｒｉは、ユーザーシステムに保持され（たとえば、サーバによってアクセスできず、または、Ｓの下で符号化されてサーバに送信される）、Ｓは、システム外部（たとえば、紙の上、または回復用に保持された別の装置内、または受託者など、他の場所にはない）に保持されたユーザの秘密である。Ｅ＿Ｓ（Ｒｉ）＝ＸｉとＸｉはＨｉで使用されたＲｉの符号化であり、Ｓはシードと呼ばれるとする。Ｈｉ，Ｘｉ　ｉ＝１、…、ｍがサーバに送信される。指標Ｈｉは次に、クライアント側で、および自分の装置内で消去される。

【００５２】

シードであるＳは、アカウントストレージ外部のユーザのメモリ（たとえば、装置または紙の上）で保持される。Ｈｉはサーバに送信されて回復検証のために保持され、ローカルコピーが消去される。サーバはさらに、侵入する攻撃者らが指標を学習することを防止するために、指標を一方向性関数でハッシュしてもよい。

【００５３】

上述の例示的な実現化例から、サーバはしたがって回答についての情報を何も受信せず、十分なエントロピーを有するハッシュ値のみを受信する。ユーザは、１つのグループと

整合するのに十分な質問に回答できるはずである。攻撃者は、１つのグループをカバーする回答さえ推測できないはずであり、Ｓへのアクセスを有さない。

【００５４】

因子は次に、認証のために使用可能である。認証プロセスまたはアカウント回復プロセスで、永続的因子を使用する試みが行なわれる。サーバは、グループのうちの１つについての質問を提示し、ユーザはグループを選択し、質問に回答し、そのＳおよびその装置を順に入力し、その秘密Ｓを使用してＸｉからＲｉを回復させる。Ｈｉは、回答グループ「Ｇｉ現在」（Gi-current：Ｇｉについての候補）を生成するユーザの現在の回答に基づいて最初から演算され、生成されたＨｉはサーバに送信される。ユーザは、回答から全指標を演算してもよく、また、これに代えて、指標における回答のうちのいくつか（例におけるＡ５など）はクリアな状態で送信可能であり、例における部分的に評価された指標Ｈ（Ｈ（Ｈ（Ｈ（Ｒ），Ｓ１）Ａ１）Ａ３）が送信可能であり、ユーザは指標演算を完了することができる（したがって、回復時のみ、いくつかの回答は隠され、いくつかの回答はオープンである）。

【００５５】

サーバは、次にハッシュされる演算された生成されたＨｉを、格納されたハッシュされたＨｉと比較する。整合がある場合、ユーザは認証される。整合がない場合（たとえば、生成されたＨｉがさまざまなグループに対して失敗した場合）、クレームした者は不合格となり、元のユーザとして認識されない。なお、代替的な例示的な一実現化例では、ユーザは、ハッシュされたバージョンＨＡＳＨ（Ｈｉ）に対してＨｉの所有を証明するプロトコルに関与していてもよい。

【００５６】

回答は、ユーザについての個人情報を必要とするかもしれないが、情報はすべて、ユーザコンピュータまたはユーザ装置にとってローカルであり、プライバシーの目的のためにサーバは情報にアクセスできず、サーバであるふりをするフィッシング第三者も情報にアクセスできない。回答は消去されるため、それらはクエリー時に再構成される。さらに、部分情報を与えることができる。たとえば、部分入れ子ハッシングが計算され、Ａ５がクリアな状態で与えられ、サーバはハッシングを完了する。

【００５７】

結合関数としてのＨ（Ｒ）＊Ｈ（Ａ１）＊Ｈ（Ａ３）＊Ｈ（Ａ５）（すなわち、十分に大きい体における個々の値のハッシュの乗算）といった一実現化例については、部分積が提供可能であり、回答のうちのいくつかはオープンであってもよく、サーバは積を完了できる。位置が固定された検査については、Ｈ（Ｒ）＊Ｈ（１，Ａ１）＊Ｈ（２，Ａ３）＊Ｈ（３，Ａ５）が提供可能であり、質問票の位置ｊでの回答Ａｉが位置ｊに関連付けられるようになっている。積は、大きい素数の順序体（large prime order field）上で行なうことができる。

【００５８】

情報は、攻撃者が回答を生成する可能性が実質的に小さくなるように、サーバの状態が与えられた個々の体を隠すのに十分なエントロピーを有するべきである。ユーザはさらに、因子が正しいサーバに与えられることを確実にする必要がある。回答のうちのいくつかをオープンにする（リアルタイム攻撃）因子または複数の因子を学習しようとすること（オフライン攻撃）が可能であり、実現化例において考慮されるべきである。

【００５９】

また、値がサーバでコミットされる（記録される）前に、高度に記憶された１組の回答を使用すべきであり、ユーザトレーニングが実施されるべきであり、これを支援するであろう。因子が永続的であることを確実にするために、思い出しにくい回答が書留められてもよい。例示的な実現化例は、汎用ソフトウェアシステムが特殊装置／リーダなどなしで採用できるといったものであってもよい。

【００６０】

例示的な実現化例は、回答が、思い出されるのではなく、外部機関からユーザによって

得られるシステムまたはプロセスを伴い得る。これらの機関は、ユーザが回答を検索できるようにするために認証に依拠しており、そのため実現化例は、上述の永続的因子を用いて暗黙の「社会的回復」を構築可能である。因子は、まず以前の検査に合格し、次に因子に埋め込まれた知識を更新することによって、徐々に構築可能である。さらに、永続的因子は、必要とされる場合に使用されるように制限されてもよく、他の因子によってサポートされてもよく、アカウントの回復またはアカウントプロセスの再クレームにおける１つの追加の決定的因子として含まれてもよい。なお、ユーザおよび受託者からの回答は、タイピング、音声、生体計測読取り、カメラなどの任意の入力方法を採用して得ることができる。

【００６１】

上述の例示的な実現化例はそれにより、ユーザがパスワードをユーザが入力する知識と置換することを可能にする。パスワードとは異なり、ユーザは、ユーザについてのいくつかの知識（それらの大部分）をたいがい知っていると思われる。パスワードの使用は、（秘密鍵のような）パスワード符号化情報のローカル復号のために利用可能である。この新しい考えを延長することが、そのような目的のためのパスワード・ドロップイン置換に使用されてもよく、それは「装置」が単なるローカル演算である別の設計の事項である。「ローカル演算」はモバイル装置上で行なわれてもよく、回答が盗まれていないことをユーザに保証するために、最終結果が、無線、ＵＳＢまたは物理的接続などのローカル通信方法を介して、コンピュータまたはサーバに送信されてもよい。

【００６２】

例示的な実現化例はまた、識別の他の手段が失われたとしても、ユーザが常に再構成できる永続的因子を伴い得る。これは、ユーザをアカウント乗取り犯と区別でき、アカウントをクレームバック（claim back）するためにユーザによって使用可能である（たとえば、質問票に基づいてクレームバックし、露出を最小限にする方法が実施される）。

【００６３】

図１（ａ）は、例示的な一実現化例に従った、装置についてのフロー図を示す。１００で、装置は、複数の質問への、ユーザによって提供された複数の応答から、複数のハッシュを生成する。提供された質問は、サーバから、または装置からであってもよく、上述のようなユーザについての個人情報に関与する質問票を利用する。

【００６４】

１０１で、装置は、複数のハッシュから認証ハッシュを生成できる。これは、認証ハッシュを生成するために複数のハッシュの多項式補間を行なうことにより、および／または、複数の質問のうちの選択されたグループに基づいて認証ハッシュを形成するために複数のハッシュのうちの１つ以上を選択することにより、実現することができる。上述の例示的な実現化例で説明されたように、ユーザは、回答すべき質問のグループを選択でき、回答はそれにより、認証ハッシュを生成するためにハッシュ可能であり、または、装置は、提供された回答の部分集合（たとえば、２つ以上）を選択し、その部分集合に基づいて認証ハッシュを生成することができる。上述の例示的な実現化例で説明されたように、秘密認証ハッシュも装置に格納でき、要件が満たされる（たとえば、質問への正しい回答のしきい値を満たす、質問の部分集合に正しく回答する、認証ハッシュが秘密認証ハッシュと整合する、など）と、セキュアプロトコルによってサーバに転送され得る。

【００６５】

装置はまた、上述のように、認証ハッシュを補間するために多項式補間を使用することで、複数のハッシュの多項式補間から認証ハッシュを生成してもよい。多項式補間アルゴリズムおよびノイズのある補間アルゴリズムといった実現化例が、採用可能である。サーバでの追加点を用いた、多項式補間のための１つ以上のエラー点、および／または多項式補間のための１つ以上の正しい点の導入を通して、しきい値を調節し、多項式補間に適用することができる。１０２で、装置は次にサーバに、生成された認証ハッシュを用いて認証することを試みる。

【００６６】

図１（ｂ）は、例示的な一実現化例に従った、回復プロセスのためのフロー図を示す。上述の例示的な実現化例で説明されたように、１０３で、装置は複数の質問を受信し、それらの中からユーザは、アカウントへのアクセスを回復するために回答すべき部分集合を選択してもよい。１０４で、提供された回答は、ユーザ装置の外部にある秘密シードから生成された乱数の使用に基づいて、認証ハッシュに変換される。１０５で、認証ハッシュはサーバに転送される。

【００６７】

図２（ａ）は、例示的な一実現化例に従った、サーバについてのフロー図を示す。２００で、サーバは、複数の個人的な質問を装置に送信してもよい。２０１で、サーバは、送信された複数の質問に応答して、装置から認証ハッシュを受信する。２０２で、サーバは次に、送信された複数の質問に応答した認証ハッシュがサーバに格納された秘密認証ハッシュと整合する場合には（ＹＥＳ）、アクセスを許可する（２０４）と決定してもよく、認証ハッシュが秘密認証ハッシュと整合しない場合には（ＮＯ）、アクセスを拒否する（２０３）と決定してもよい。認証ハッシュは、複数のハッシュの多項式補間、および、複数の質問のうちの選択されたグループに基づいて認証ハッシュを形成するための、複数のハッシュのうちの１つ以上の選択から生成されてもよい。実現化例に依存して、サーバは、秘密認証ハッシュおよびしきい値に基づいて、ノイズのある補間アルゴリズムで使用するための１つ以上のエラー点および１つ以上の正しい点を送信してもよい。サーバはまた、ハッシュが格納された秘密ハッシュと整合するかどうか判断するために、受信された認証ハッシュのハッシングを行なってもよい。

【００６８】

別の例では、サーバは、送信された質問のうちの選択されたグループに基づいて、複数の秘密認証ハッシュから秘密認証ハッシュを選択してもよく、複数の秘密認証ハッシュの各々は、複数の質問のうちの少なくとも２つに関連付けられている。送信された質問のうちの選択されたグループは、装置で、またはサーバによって選択されてもよい。これはたとえば、上述のような回復プロセスで実現可能である。

【００６９】

図２（ｂ）は、例示的な一実現化例に従った、サーバからの回復プロセスのためのフロー図を示す。２０５で、サーバは、複数の個人的な質問を装置に送信してもよい。２０６で、サーバは、送信された複数の質問に応答して、装置から認証ハッシュを受信する。認証ハッシュは潜在的に、上述の例示的な実現化例で説明されたような指標のうちの１つである。２０７で、サーバは次に、認証ハッシュがサーバに格納された指標のうちの１つと整合する場合には（ＹＥＳ）、ユーザーアカウントを回復するプロセスを開始する（２０９）と決定してもよく、認証ハッシュが格納された指標のどれとも整合しない場合には（ＮＯ）、アクセスを拒否する（２０８）と決定してもよい。

【００７０】

例示的な処理環境

図３は、いくつかの例示的な実現化例で使用するのに好適な例示的なコンピューティング装置を有する、例示的なコンピューティング環境を示す。コンピューティング環境３００におけるコンピューティング装置３０５は、１つ以上の処理部、コア、またはプロセッサ３１０、メモリ３１５（たとえば、ＲＡＭ、ＲＯＭなど）、内部ストレージ３２０（たとえば、磁気、光学、固体ストレージ、および／または有機）、および／またはＩ／Ｏインターフェイス３２５を含んでいてもよく、それらのいずれも、情報を通信するための通信機構またはバス３３０上に結合可能であり、または、コンピューティング装置３０５に埋込み可能である。

【００７１】

コンピューティング装置３０５は、入力／ユーザインターフェイス３３５および出力装置／インターフェイス３４０に通信可能に結合可能である。入力／ユーザインターフェイス３３５および出力装置／インターフェイス３４０のいずれか一方または双方は、有線または無線インターフェイスであってもよく、取り外し可能であってもよい。入力／ユーザ

インターフェイス３３５は、入力を提供するために使用可能な、物理的なまたは仮想の任意の装置、コンポーネント、センサ、またはインターフェイス（たとえば、ボタン、タッチスクリーン・インターフェイス、キーボード、ポインティング／カーソル制御装置、マイク、カメラ、点字、運動センサ、光学式読取り装置など）を含んでいてもよい。出力装置／インターフェイス３４０は、ディスプレイ、テレビ、モニタ、プリンタ、スピーカ、点字などを含んでいてもよい。いくつかの例示的な実現化例では、入力／ユーザインターフェイス３３５および出力装置／インターフェイス３４０は、コンピューティング装置３０５に埋込可能であり、または物理的に結合可能である。他の例示的な実現化例では、他のコンピューティング装置は、コンピューティング装置３０５のために、入力／ユーザインターフェイス３３５および出力装置／インターフェイス３４０として機能し、またはそれらの機能を提供してもよい。

【００７２】

コンピューティング装置３０５の例は、高度モバイル装置（たとえば、スマートフォン、車両および他のマシンにおける装置、人間および動物によって運ばれる装置など）、モバイル装置（たとえば、タブレット、ノートブック、ラップトップ、パーソナルコンピュータ、ポータブルテレビ、ラジオなど）、および移動性のために設計されていない装置（たとえば、デスクトップコンピュータ、他のコンピュータ、情報キオスク、１つ以上のプロセッサが埋め込まれた、および／または結合されたテレビ、ラジオ、サーバなど）を含み得るが、それらに限定されない。

【００７３】

コンピューティング装置３０５は、同じまたは異なる構成の１つ以上のコンピューティング装置を含む、任意の数のネットワーク化されたコンポーネント、装置、およびシステムと通信するために、（たとえばＩ／Ｏインターフェイス３２５を介して）外部ストレージ３４５およびネットワーク３５０に通信可能に結合可能である。コンピューティング装置３０５または任意の接続されたコンピューティング装置は、サーバ、クライアント、シンサーバ、汎用マシン、特殊用途マシン、または別のラベルとして機能し、それらのサービスを提供し、またはそのように呼ばれてもよい。

【００７４】

Ｉ／Ｏインターフェイス３２５は、少なくとも、コンピューティング環境３００におけるすべての接続されたコンポーネント、装置、およびネットワークとの間で情報を通信するために、任意の通信またはＩ／Ｏプロトコルもしくは規格（たとえば、イーサネット（登録商標）、８０２．１１ｘ、ユニバーサル・システム・バス、ＷｉＭａｘ、モデム、セルラー・ネットワーク・プロトコルなど）を用いる有線および／または無線インターフェイスを含み得るが、それらに限定されない。ネットワーク３５０は、任意のネットワーク、またはネットワークの組合せであってもよい（たとえば、インターネット、ローカルエリアネットワーク、ワイドエリアネットワーク、電話ネットワーク、セルラーネットワーク、衛星ネットワークなど）。

【００７５】

コンピューティング装置３０５は、信号媒体および記憶媒体を含む、コンピュータ使用可能媒体またはコンピュータ読取可能媒体を用いて、使用および／または通信できる。信号媒体は、伝送媒体（たとえば、金属ケーブル、光ファイバー）、信号、搬送波などを含む。記憶媒体は、磁気媒体（たとえば、ディスクおよびテープ）、光学媒体（たとえば、ＣＤ　ＲＯＭ、デジタルビデオティスク、ブルーレイディスク）、固体媒体（たとえば、ＲＡＭ、ＲＯＭ、フラッシュメモリ、固体ストレージ）、および他の不揮発性ストレージまたはメモリを含む。

【００７６】

コンピューティング装置３０５は、いくつかの例示的なコンピューティング環境において、手法、方法、アプリケーション、プロセス、またはコンピュータ実行可能命令を実現するために使用可能である。コンピュータ実行可能命令は、一時的媒体から検索可能であり、また、非一時的媒体に格納されてそこから検索されることが可能である。実行可能命

令は、任意のプログラミング言語、スクリプト言語、およびマシン語（たとえば、Ｃ、Ｃ＋＋、Ｃ＃、Ｊａｖａ（登録商標）、ビジュアル・ベーシック、パイソン、パール、ＪａｖａＳｃｒｉｐｔ（登録商標）など）のうちの１つ以上から生じ得る。

【００７７】

プロセッサ３１０は、自然または仮想環境において、任意のオペレーティングシステム（ＯＳ）（図示せず）の下で実行可能である。１つ以上のアプリケーションを配備することができ、それは、論理ユニット３６０と、アプリケーション・プログラミング・インターフェイス（ＡＰＩ）ユニット３６５と、入力ユニット３７０と、出力ユニット３７５と、認証ユニット３８０と、回復ユニット３８５と、乱数発生器ユニット３９０と、異なるユニットが互いに、ＯＳと、および他のアプリケーション（図示せず）と通信するためのユニット間通信機構３９５とを含む。たとえば、装置またはサーバとしての実現化例に依存して、認証ユニット３８０、回復ユニット３８５、および乱数発生器ユニット３９０は、図１（ａ）、図１（ｂ）、図２（ａ）および図２（ｂ）に示すような１つ以上のプロセスを実現してもよい。回復ユニット３８５はまた、図１（ｂ）および図２（ｂ）の上述の例示的な実現化例で説明されたような回復プロセスを実現してもよい。説明されたユニットおよび要素は、設計、機能、構成、または実現化例の点で変更可能であり、提供された説明に限定されない。

【００７８】

いくつかの例示的な実現化例では、情報または実行命令がＡＰＩユニット３６５によって受信されると、それは、１つ以上の他のユニット（たとえば、論理ユニット３６０、入力ユニット３７０、出力ユニット３７５、認証ユニット３８０、回復ユニット３８５、および乱数発生器ユニット３９０）に通信されてもよい。たとえば、乱数発生器ユニット３９０は、ハッシュを生成するかまたは提出用の質問を選択し、ＡＰＩユニット３６５を使用して認証ユニット３８０および回復ユニット３８５と通信し、上述の例示的な実現化例で説明されているように乱数を提供するために使用されてもよい。認証ユニット３８０は、認証ハッシュを格納された秘密認証ハッシュと比較するために、ＡＰＩユニット３６５を介して回復ユニット３８５と相互作用してもよい。

【００７９】

場合によっては、論理ユニット３６０は、上述のいくつかの例示的な実現化例において、ユニット間の情報フローを制御して、ＡＰＩユニット３６５、入力ユニット３７０、出力ユニット３７５、認証ユニット３８０、回復ユニット３８５、および乱数発生器ユニット３９０によって提供されるサービスを指示するように構成されてもよい。たとえば、１つ以上のプロセスまたは実現化例のフローは、論理ユニット３６０のみによって、またはＡＰＩユニット３６５とともに制御されてもよい。

【００８０】

例示的な処理環境

図４は、いくつかの例示的な実施形態が実現され得る例示的なオンライン環境を示す。環境４００は装置４０５～４４５を含み、各々は、たとえばネットワーク４５０を介して、少なくとも１つの他の装置に通信可能に接続されている。いくつかの装置は、（たとえば、装置４２５を介して）１つ以上の記憶装置４３０および４４５に通信可能に接続されてもよい。

【００８１】

１つ以上の装置４０５～４４５の一例は、図３で説明するコンピューティング装置であってもよい。装置４０５～４４５は、コンピュータ４２５（たとえば、個人用または業務用）、車両４２０に関連付けられた装置、モバイル装置４１０（たとえば、スマートフォン）、テレビ４１５、モバイルコンピュータ４０５、サーバコンピュータ４５０、コンピューティング装置４３５～４４０、記憶装置４３０、４４５を含み得るが、それらに限定されない。装置４０５～４４５のうちのいずれも、環境４００において図示された１つ以上の装置、および／または環境４００において図示されていない装置からの１つ以上のサービスにアクセスしてもよく、および／または、そのような装置に１つ以上のサービスを

提供してもよい。装置間のアクセスは、有線、無線で行なわれてもよく、ユーザ音声やカメラ画像のようなマルチメディア通信によるものであってもよい。

【００８２】

ユーザは、ネットワーク４５０を介して例示的な実現化例を実現するために、上に説明されたように装置を制御してもよい。例示的な実現化例に関連付けられた情報は、たとえば記憶装置４３０または４４５にそれぞれ格納されてもよい。

【００８３】

ここに説明されたシステムが、ユーザについての個人情報を収集する、または個人情報を利用し得る状況では、ユーザには、プログラムまたは機能がユーザ情報（たとえば、ユーザの社会的ネットワーク、社会的行動または活動、職業、ユーザの好み、またはユーザの現在位置についての情報）を収集するかどうかを制御する機会、もしくは、ユーザにより関連し得るコンテンツサーバからコンテンツを受信するかどうか、および／またはどのように受信するかを制御する機会が提供されてもよい。加えて、或るデータは、個人を識別可能な情報が除去されるように、格納または使用される前に１つ以上のやり方で処理されてもよい。たとえば、ユーザについて、個人を識別可能な情報がまったく判断できないように、ユーザの身元が処理されてもよく、または、（市、郵便番号、または国家レベルなどまで）位置情報が得られる場合、ユーザの特定の位置が判断できないように、ユーザの地理的な位置が一般化されてもよい。このため、ユーザは、ユーザについて情報がどのように収集され、コンテンツサーバによってどのように使用されるか、に対する制御を有していてもよい。

【００８４】

いくつかの例示的な実現化例を示し、説明してきたが、これらの例示的な実現化例は、ここに説明された主題を、この分野の当業者に伝えるために提供されている。ここに説明された主題は、説明された例示的な実現化例に限定されることなく、さまざまな形で実現され得る、ということが理解されるべきである。ここに説明された主題は、それらの具体的に定義または説明された事項がなくても、もしくは、説明されていない他のまたは異なる要素または事項があっても、実践可能である。これらの例示的な実現化例において、添付された請求項およびそれらの均等物で定義されるような、ここに説明された主題から逸脱することなく、変更を行なってもよい、ということは、この分野の当業者によって理解されるであろう。

【図１（ａ）】

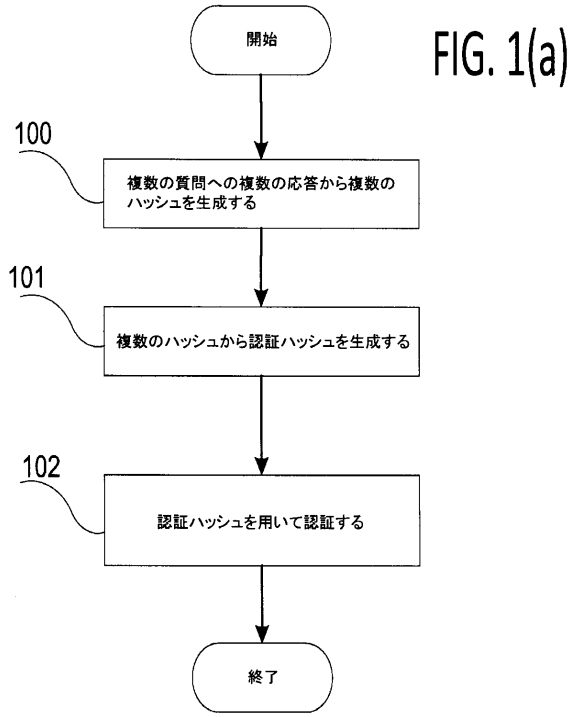

FIG. 1(a)

【図１（ｂ）】

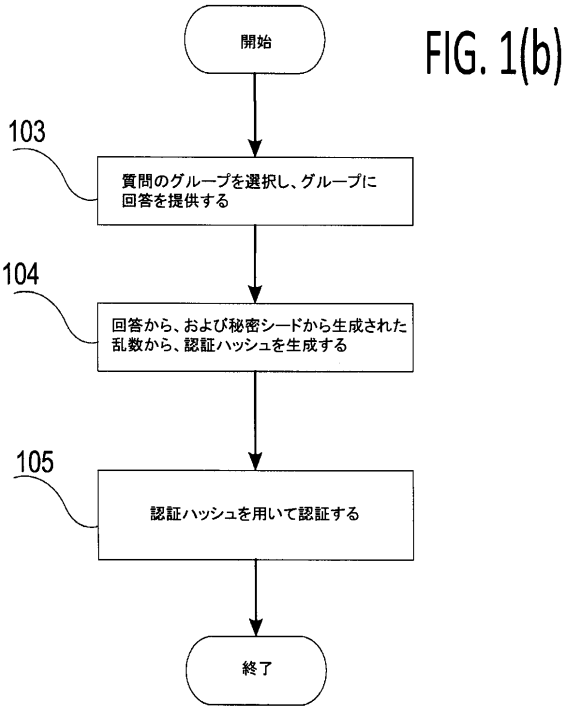

FIG. 1(b)

【図２（ａ）】

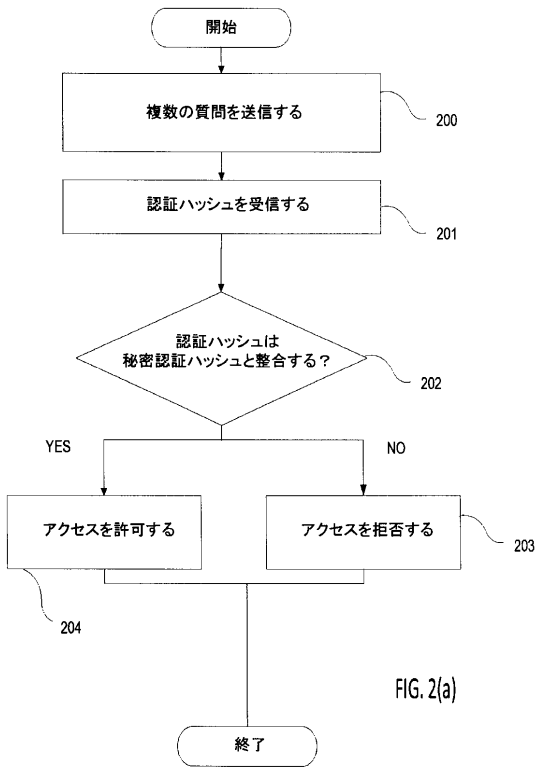

FIG. 2(a)

【図２（ｂ）】

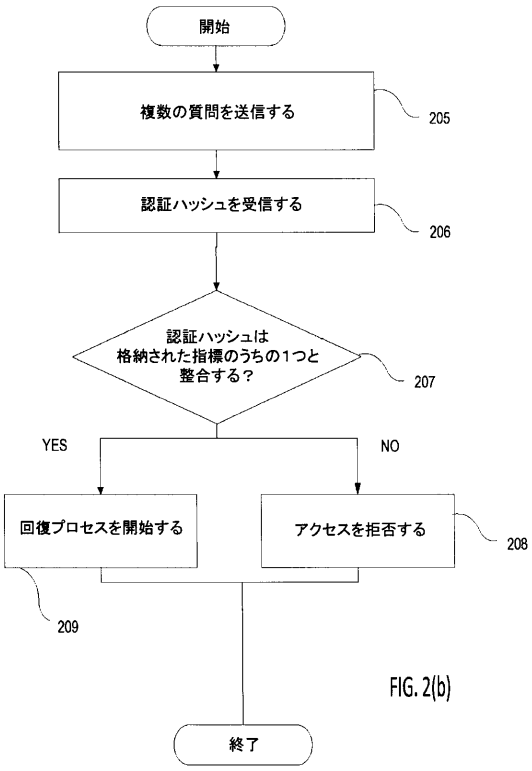

FIG. 2(b)

【図３】

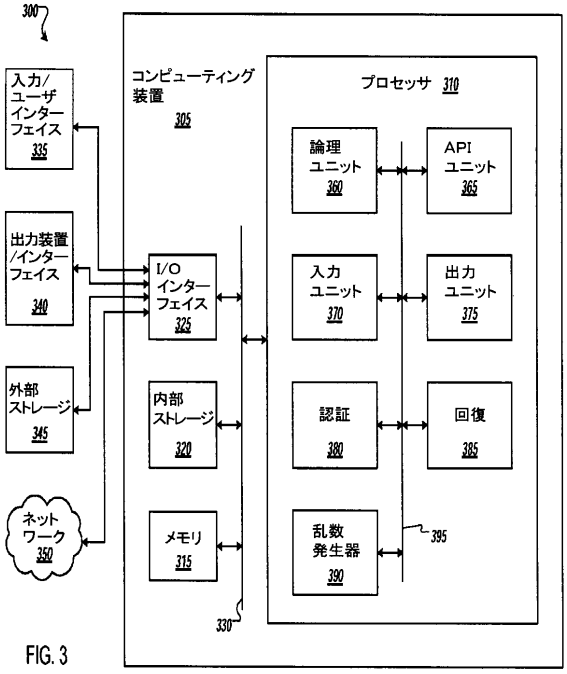

FIG. 3

【図４】

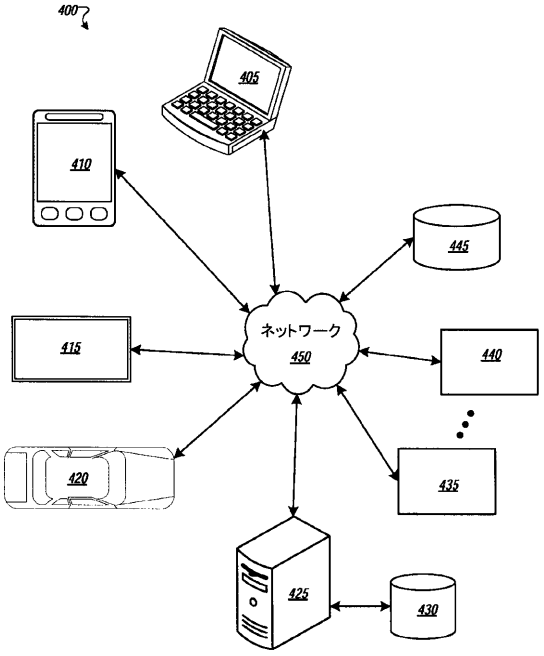

FIG. 4

フロントページの続き

合議体
審判長　高木　進
審判官　石井　茂和
審判官　山崎　慎一

(56)参考文献　米国特許第４，６３３，４７０（US，A）
R. J. McEliece, D. V. Sarwate「On sharing secrets and Reed−Solomon codes」Communications of the ACM Volume24 Issue9, Sept. 1981 pages583−584
Moni Naor, Benny Pinkas「Oblivious transfer and polynomial evaluation」STOC'99 Proceedings of the thirty−first annual ACM symposium on Theory of computing May 01−04, 1999 pages245−254
Cristophe Tartary, Huaxiong Wang「Dynamic Threshold and Cheater Resistance for Shamir Secret Sharing Scheme」Inscrypt2006 Beijing Chaina November29−December1, 2006 pp103−117
Daniel Bleichenbacher, Phong Q. Nguyen「Noisy Polynomial Interpolation and Noisy Chinese Remaindering」EUROCRYPT2000 12 May, 2000 pp53−69
平野　亮　他，パスワード運用管理に関する考察および提案とその開発，電子情報通信学会技術研究報告，日本，社団法人電子情報通信学会，２０１１年１１月　７日，Ｖｏｌ．１１１，Ｎｏ．２８５，ｐ．１２９−１３４
D. W. Davies and W. L. Price／上園　忠弘，ネットワーク・セキュリティ，日本，日経マグロウヒル社，１９８５年１２月　５日，１版１刷，ｐ．１２６−１３１

(58)調査した分野(Int.Cl.，ＤＢ名)
H04L 9/00 675C, G06F 21/31