(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2016-4580
(P2016-4580A)
(43)公開日　平成28年1月12日(2016.1.12)

(51)Int.Cl.	F I	テーマコード（参考）
G06F 21/46 (2013.01)	G06F 21/46	

審査請求 未請求　請求項の数 14　ＯＬ　外国語出願　（全 14 頁）

(21)出願番号　特願2015-111264 (P2015-111264)
(22)出願日　平成27年6月1日 (2015.6.1)
(31)優先権主張番号　14305884.0
(32)優先日　平成26年6月12日 (2014.6.12)
(33)優先権主張国　欧州特許庁 (EP)

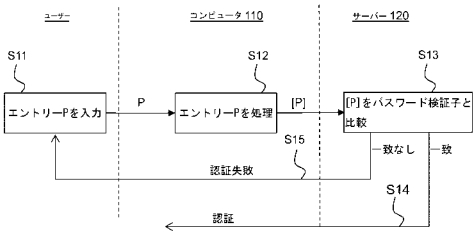
（特許庁注：以下のものは登録商標）
１．ＴＥＣＨＮＩＣＯＬＯＲ

(71)出願人　501263810
トムソン　ライセンシング
Ｔｈｏｍｓｏｎ　Ｌｉｃｅｎｓｉｎｇ
フランス国，　９２１３０　イツシー　レ　ムーリノー，　ル　ジヤンヌ　ダルク，　１－５
１－５，　ｒｕｅ　Ｊｅａｎｎｅ　ｄ’Ａｒｃ，　９２１３０　ＩＳＳＹ　ＬＥＳ　ＭＯＵＬＩＮＥＡＵＸ，　Ｆｒａｎｃｅ
(74)代理人　100107766
弁理士　伊東　忠重
(74)代理人　100070150
弁理士　伊東　忠彦
(74)代理人　100091214
弁理士　大貫　進介

最終頁に続く

(54)【発明の名称】パスワード認証のための装置および方法

(57)【要約】　（修正有）
【課題】ミスタイプされたパスワードを許容する認証システムを提供する。
【解決手段】ユーザーはコンピュータ１１０にエントリーＰを入力Ｓ１１する。コンピュータ１１０はエントリーＰを処理Ｓ１２して複数のサブエントリー[Ｐ]を得る。各サブエントリーは、エントリーＰをパディングして固定長のエントリーとし、固定長のエントリーからk個のキャラクタの種々の組み合わせが欠けているいくつかのストリングを生成し、一方向性関数に通して得られる。サーバー１２０はそれらのサブエントリー[Ｐ]を受領し、各サブエントリーをユーザーのパスワード検証子と比較Ｓ１３し、少なくとも一つのサブエントリーがパスワード検証子と一致すればユーザーは認証され、通知がコンピュータ１１０を介してユーザーに送られる。
【選択図】図２



【特許請求の範囲】
【請求項１】
パスワード・エントリーを処理する装置であって、
・キャラクタ列を含むパスワード・エントリーを受領する手段と、
・前記パスワード・エントリーから複数のサブエントリーを生成する手段であって、各サブエントリーは前記キャラクタ列よりk個のキャラクタの異なる組み合わせを除いたものに等しく、kは整数である、手段と、
・各サブエントリーに対して関数をそれぞれ使うことによって、パスワード検証子を生成する手段とを有する、
装置。
【請求項２】
前記関数が一方向性ハッシュ関数または暗号化関数である、請求項１記載の装置。
【請求項３】
前記パスワード・エントリーをパディングして長さLのパディングされたパスワード・エントリーを得る手段をさらに有しており、Lは整数であり、前記複数のサブエントリーを生成する手段は、前記パディングされたパスワード・エントリーから前記サブエントリーを生成するためのものである、請求項１記載の装置。
【請求項４】
当該装置はユーザー・デバイスであり、前記受領する手段はユーザー・インターフェースであり、前記ユーザー・デバイスはさらに、前記パスワード検証子を認証装置に出力するよう構成されている通信インターフェースを有する、請求項１記載の装置。
【請求項５】
パスワード・エントリーを処理する方法であって、
・キャラクタ列を含むパスワード・エントリーをインターフェースによって受領する段階と、
・前記パスワード・エントリーから複数のサブエントリーをプロセッサによって生成する段階であって、各サブエントリーは前記キャラクタ列よりk個のキャラクタの異なる組み合わせを除いたものに等しく、kは整数である、段階と、
・各サブエントリーに対して関数をそれぞれ使うことによって、前記プロセッサによってパスワード検証子を生成する段階とを含む、
方法。
【請求項６】
ユーザーによって入力されたパスワード・エントリーの認証のための装置であって、当該装置は、
・Mが整数であるとして、あるユーザーについてのM個のパスワード検証子を記憶する手段と、
・Nが整数であるとして、前記パスワード・エントリーを表わすN個の得られたパスワード検証子を得る手段と、
・前記得られたパスワード検証子のそれぞれを、前記記憶されているパスワード検証子と比較する手段と、
・少なくとも一つの得られたパスワード検証子が記憶されているパスワード検証子に一致するとの判定に際して前記パスワード・エントリーを認証する手段とを有する、
装置。
【請求項７】
ユーザー・デバイスから前記得られたパスワード検証子を受領する手段と、前記得られたパスワード検証子を前記プロセッサに送る手段とをさらに有する、請求項６記載の装置。
【請求項８】
ユーザー・デバイスから前記パスワード・エントリーを受領する手段と、前記パスワード・エントリーを前記プロセッサに送る手段と、前記記憶されたパスワード検証子を生成

するために関数を使う手段とを有する、請求項６記載の装置。
【請求項９】
前記記憶されたパスワード検証子および前記得られたパスワード検証子は順序付けられており、前記比較する手段は、各得られたパスワード検証子を同じ順序をもつ前記記憶されたパスワード検証子とのみ比較するよう構成されている、請求項６記載の装置。
【請求項１０】
ユーザーによって入力されたパスワード・エントリーの認証のための方法であって、
・前記パスワード・エントリーを表わすN個の得られたパスワード検証子をプロセッサによって得る段階と、
・前記得られたパスワード検証子を、M個の記憶されているパスワード検証子と前記プロセッサによって比較する段階と、
・少なくとも一つの得られたパスワード検証子が記憶されているパスワード検証子に一致するとの判定に際して前記パスワード・エントリーを前記プロセッサによって認証する段階とを含む、
方法。
【請求項１１】
請求項５記載の方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含むコンピュータ・プログラム。
【請求項１２】
請求項５記載の方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含み、非一時的なコンピュータ可読媒体上に記憶されているコンピュータ・プログラム・プロダクト。
【請求項１３】
請求項１０記載の方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含むコンピュータ・プログラム。
【請求項１４】
請求項１０記載の方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含み、非一時的なコンピュータ可読媒体上に記憶されているコンピュータ・プログラム・プロダクト。
【発明の詳細な説明】
【技術分野】
【０００１】
本開示は概括的にはコンピュータ・システムに、詳細にはそのようなシステムにおけるパスワードの扱いに関する。
【背景技術】
【０００２】
本節は、以下で記述されるおよび／または特許請求される本開示のさまざまな側面に関係しうる技術のさまざまな側面を読者に紹介するために意図されたものである。この議論は、本開示のさまざまな側面のよりよい理解を容易にするために、読者に背景情報を提供する助けとなると信ずる。よって、これらの陳述は、従来技術の自認ではなく、この観点で読まれるべきであることは理解しておくべきである。
【０００３】
パスワードは今日のコンピュータ・システムにおいて、たとえばログオンのためにユーザーを認証するために、いたるところで使われている。その一般的な定義では、パスワードは、あらかじめ定義されたアルファベットの範囲内で取られる記号（「キャラクタ」）の連続から構成される（たとえば暗証番号のための4つの数値）。パスワードは一般に、長いほど、特にパスワードが大文字および小文字、数字および&、"および#のような特殊キャラクタの混合である場合に、より強くなる。しかしながら、より複雑なパスワードは一般に、タイプするのがより複雑である。タイプされるキャラクタが画面上に現われないので特にそうである。さらに、ユーザーは一日に何回もパスワードを入力することを余儀

なくされることがあるので、パスワードはしばしば非常に素速くタイプされる。そのため、入力されたパスワードがタイプ誤りを含むことがあることは驚くには当たらない。さらに、スマートフォンおよびタブレットのようなタッチスクリーン・ベースの装置は、パスワードを含め何らかのテキストを入力するために仮想キーボードを使う。この型の入力では、タイプ誤りはきわめて頻繁に起こりうる。

【０００４】

従来技術は、タイプ誤りに耐性のあるパスワードを提供するいくつかの解決策を含んでいる。

【０００５】

パスワード・バリエーター（Password Variator）というツールは、打ちそこなったキャラクタ、ダブったキャラクタ、余計なキャラクタ、誤った順序および大文字／小文字の誤りといった三つまでのタイプミスをエミュレートして、パスワードに対するあらゆる可能な変形をもつファイルを構築する。Andrew MehlerおよびSteven Skienaは非特許文献１において異なる解決策を提供している。彼らの解決策は、ハッシュする前にパスワードを処理する――たとえば入力されたパスワードをアルファベット順にソートして入れ替わりを訂正する――ことで、わずかに綴りを誤ったパスワードが正しいパスワードと同じハッシュ値にハッシュされる可能性が高くなるようにする。一つまたは二つのタイプミスよりずっと大きく異なるパスワードを含め多くの異なるパスワードがハッシュされると同じ値になるので、「受け容れ可能な」パスワードのそのような増倍がパスワードの強さを大きく低下させることは明らかである。

【０００６】

特許文献１および特許文献２の解決策は、入力されたパスワードを記憶されているパスワードと「パスワード空間」において、すなわち平文において比較して、前者が後者と「同様」であるかどうかを判定する。しかしながら、これらの解決策はパスワードの平文バージョンの比較を必要とするので、平文でのパスワードの記憶が明らかなセキュリティ上の理由により一般に受け容れられないことである現実のシステムにおいては、使用できない。一般に、認証は、ユーザーによってタイプされたパスワード・エントリーのハッシュされたバージョンと比較されるべき、パスワードのハッシュされたバージョンを記憶しているサーバーによって扱われる。これは、パスワード・ファイルを盗むことの価値を減じさせる。

【０００７】

特許文献３は、中でも、ミスタイプされたパスワードが入力される回数の計数を続け、回数が十分大きい、たとえば10回の場合にミスタイプされたパスワードを受け容れ可能なパスワードとして記憶する機能を提供するPCのような装置を教示している。当業者は、この解決策は、ミスタイプされたパスワードの追加に対していかなる制御もないようなので、安全でないことを理解するであろう。誤ったパスワードが10回入力されたら、それが受け容れ可能なパスワードとして記憶される。つまりハッカーはパスワードを10回入力しさえすれば、それが受け容れられるようになるのである。

【先行技術文献】

【特許文献】

【０００８】

【特許文献１】米国特許第7373516号
【特許文献２】特開2005-208763
【特許文献３】特開2007-114976
【特許文献４】国際公開第2006/048043号

【非特許文献】

【０００９】

【非特許文献１】Andrew Mehler and Steven Skiena、Improving Usability Through Password-Corrective Hashing［パスワード訂正ハッシングを通じた使いやすさの改善］

【発明の概要】
【発明が解決しようとする課題】
【００１０】
したがって、従来技術の解決策の欠点なしにミスタイプされたパスワードを許容する認証システムを許容できる解決策が必要とされている。
【課題を解決するための手段】
【００１１】
第一の側面では、本願の原理は、パスワード・エントリーを処理する装置であって、キャラクタ列を含むパスワード・エントリーを受領する手段と、前記パスワード・エントリーから複数のサブエントリーを生成する手段であって、各サブエントリーは前記キャラクタ列よりk個のキャラクタの異なる組み合わせを除いたものに等しく、kは整数である、手段と、各サブエントリーに対して関数をそれぞれ使うことによって、パスワード検証子を生成する手段とを有する装置に向けられる。
【００１２】
第一の実施形態では、前記関数は一方向性ハッシュ関数または暗号化関数である。
【００１３】
第二の実施形態では、当該装置は、前記パスワード・エントリーをパディングして長さLのパディングされたパスワード・エントリーを得る手段をさらに有しており、Lは整数であり、前記複数のサブエントリーを生成する手段は、前記パディングされたパスワード・エントリーから前記サブエントリーを生成するためのものである。
【００１４】
第三の実施形態では、当該装置は、ユーザー・デバイスであり、前記受領する手段はユーザー・インターフェースであり、前記ユーザー・デバイスはさらに、前記パスワード検証子を認証装置に出力するよう構成されている通信インターフェースを有する。
【００１５】
第二の側面では、本願の原理は、パスワード・エントリーを処理する方法であって、キャラクタ列を含むパスワード・エントリーをインターフェースによって受領する段階と、前記パスワード・エントリーから複数のサブエントリーをプロセッサによって生成する段階であって、各サブエントリーは前記キャラクタ列よりk個のキャラクタの異なる組み合わせを除いたものに等しく、kは整数である、段階と、各サブエントリーに対して関数をそれぞれ使うことによって、前記プロセッサによってパスワード検証子を生成する段階とを含む方法に向けられる。
【００１６】
第三の側面では、本願の原理は、ユーザーによって入力されたパスワード・エントリーの認証のための装置であって、当該装置は、Mが整数であるとして、あるユーザーについてのM個のパスワード検証子を記憶する手段と、Nが整数であるとして、前記パスワード・エントリーを表わすN個の得られたパスワード検証子を得る手段と、前記得られたパスワード検証子のそれぞれを、前記記憶されているパスワード検証子と比較する手段と、少なくとも一つの得られたパスワード検証子が記憶されているパスワード検証子に一致するとの判定に際して前記パスワード・エントリーを認証する手段とを有する、装置に向けられる。
【００１７】
第一の実施形態では、当該装置はさらに、ユーザー・デバイスから前記得られたパスワード検証子を受領する手段と、前記得られたパスワード検証子を前記プロセッサに送る手段とを有する。
【００１８】
第二の実施形態では、当該装置はさらに、ユーザー・デバイスから前記パスワード・エントリーを受領する手段と、前記パスワード・エントリーを前記プロセッサに送る手段と、前記記憶されたパスワード検証子を生成するために関数を使う手段とを有する。
【００１９】

第三の実施形態では、前記記憶されたパスワード検証子および前記得られたパスワード検証子は順序付けられており、前記比較する手段は、各得られたパスワード検証子を同じ順序をもつ前記記憶されたパスワード検証子とのみ比較するよう構成される。

【００２０】

第四の側面では、本願の原理は、ユーザーによって入力されたパスワード・エントリーの認証のための方法であって、前記パスワード・エントリーを表わすN個の得られたパスワード検証子をプロセッサによって得る段階と、前記得られたパスワード検証子を、M個の記憶されているパスワード検証子と前記プロセッサによって比較する段階と、少なくとも一つの得られたパスワード検証子が記憶されているパスワード検証子に一致するとの判定に際して前記パスワード・エントリーを前記プロセッサによって認証する段階とを含む、方法に向けられる。

【００２１】

第五の側面では、本願の原理は、前記第二の側面に基づく方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含むコンピュータ・プログラムに向けられる。

【００２２】

第六の側面では、本願の原理は、前記第二の側面に基づく方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含み、非一時的なコンピュータ可読媒体上に記憶されているコンピュータ・プログラム・プロダクトに向けられる。

【００２３】

第七の側面では、本願の原理は、前記第四の側面に基づく方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含むコンピュータ・プログラムに向けられる。

【００２４】

第八の側面では、本願の原理は、前記第四の側面に基づく方法の段階を実装するためにプロセッサによって実行可能なプログラム・コード命令を含み、非一時的なコンピュータ可読媒体上に記憶されているコンピュータ・プログラム・プロダクトに向けられる。

【図面の簡単な説明】

【００２５】

本開示の好ましい事項についてこれから、限定しない例として、付属の図面を参照して説明する。

【図１】本開示が実装されうる例示的なシステムを示す図である。

【図２】本開示のパスワード認証の例示的な方法を示す図である。

【発明を実施するための形態】

【００２６】

図１は、本開示が実装されうる例示的なシステムを示している。本システムは、コンピューティング・デバイス（「コンピュータ」）１１０および認証サーバー１２０を有する。コンピュータ１１０および認証サーバー（以下「サーバー」）１２０は、標準的なパーソナル・コンピュータ（PC）またはワークステーションのような、計算を実行できるいかなる種類の好適なコンピュータまたはデバイスであってもよい。コンピュータ１１０およびサーバー１２０はそれぞれ好ましくは、少なくとも一つのプロセッサ１１１、１２１と、パスワード検証子を記憶するための内部、外部RAMメモリ１１２、１２２と、ユーザーとのインターフェースをもつためのユーザー・インターフェース１１３と、接続１３０を通じた他のデバイスとの対話のための第二のインターフェース１１４、１２４とを有する。コンピュータ１１０およびサーバー１２０はそれぞれ好ましくは、プロセッサによって実行されたときに以下に記述されるパスワード方法のいずれかを実行する命令を記憶している非一時的デジタル・データ担体１４０、１５０からのソフトウェア・プログラムを読むためのインターフェースをも有する。当業者は、図示したデバイスが明快の理由により非常に単純化されており、現実のデバイスはさらに持続的記憶のような事項を有することを理解するであろう。当業者は、パスワードが単に当該コンピュータ自身へのアクセスを

提供するだけである場合には本開示が単にコンピュータ１１０上で実装されてもよいことを理解するであろう。その場合、サーバー１２０は当該コンピュータにおいて実装される。

【００２７】

図２は、本開示のパスワード認証の第一の例示的な方法を示している。ユーザーがすでにユーザー名および基準パスワードを有していることが想定される。一人のユーザーしかいないときなどユーザー名が黙示されている状況では、ユーザー名は余計であることを注意しておく。スマートフォンやタブレットではそうでありうる。サーバー１２０が、必要とされる場合にユーザー名と、基準パスワードから「耐性」関数によって生成された複数のパスワード検証子とを記憶することも想定される。耐性関数の逆を求めることは計算上困難であることが好ましい。つまり、値xおよび耐性関数f()が与えられたとき、f(x)を計算することは簡単だが、値y＝f(x)を与えられたとき、xの値または同じ結果を生じる異なる値を見出すことは計算上困難である。サーバー１２０が基準パスワードから前記複数のパスワード検証子を生成することが可能であるが、ユーザー・デバイス１１０（またはユーザー入力が新たなパスワードを生成する別のデバイス）がパスワード検証子を生成して、好ましくは認証された該パスワード検証子を、サーバー１２０に記憶のために送ることが好ましい。

【００２８】

前記複数のパスワード検証子は、ひとたび基準パスワードが生成されたら、システムを、パスワードを入力するときにユーザーが犯す誤りに対して耐性にすることが意図されている。該パスワード検証子は、これから述べるような種々の仕方で生成できる：

・パディング：耐性関数は、さらなる処理の前に基準パスワードにパディングしてある長さにすることができる。

・受け容れられることのできるタイプ誤りの最大数：耐性関数は入力パスワードにおける1個、2個、3個……までのタイプ誤りを受け容れるよう意図されたパスワード検証子を生成できる。

・サブ関数：ハッシュ関数（SHA-1、SHA-3、MD5……）または暗号化関数（RSA、楕円曲線暗号）のような種々のサブ関数が――パディングされてまたはパディングされずに――使用されてもよい。

【００２９】

以下の例解用の例は、TECHNICOLORに等しい基準パスワード、12キャラクタへのパディング、1個のタイプ誤りの最大受け容れおよびh()で表わされるハッシュ関数を使う。

【００３０】

耐性関数はまず、いくつかのキャラクタ、たとえばアステリスクを追加することによって、基準パスワードを12キャラクタにパディングする：TECHNICOLOR*。

【００３１】

1個までのタイプ誤りを受け容れるために、耐性関数は次いで、12（パディングされた長さに等しい値）通りのサブパスワードを生成する。ここで、各サブパスワードは、パディングされた基準パスワード中のキャラクタの一つを省いたものである：

ECHNICOLOR*
TCHNICOLOR*
TEHNICOLOR*
TECNICOLOR*
TECHICOLOR*
TECHNCOLOR*
TECHNIOLOR*
TECHNICLOR*
TECHNICOOR*
TECHNICOLR*
TECHNICOLO*

TECHNICOLOR
。
【００３２】
耐性関数は最後に、好ましくはソルト（salt）と組み合わされた各サブパスワードに対してサブ関数を使用して、12個のパスワード検証子、h(ECHNICOLOR*)、h(TCHNICOLOR*)……を生成する。異なるサブパスワードについて異なるサブ関数が使用されてもよい。耐性関数は、修正されない形の、パディングされた形のまたは両方の形の基準パスワードに対してサブ関数を使用することによってさらなるパスワード検証子を生成できることが理解されるであろう。
【００３３】
別の言い方をすれば、本耐性方法は、L個のキャラクタp[1]‖p[2]‖……‖p[L]をもつパディングされたまたはパディングされない基準パスワード（reference password）RPを取り、サブパスワードi中でキャラクタiを省き、それぞれの好ましくはソルトを加えられたサブパスワードにサブ関数Hを適用することによってL個のサブパスワードを生成する。その結果、L個のパスワード検証子が生じる：
$RP_1=H_1(p[2]‖p[3]‖…‖p[L]‖salt)$
$RP_2=H_2(p[1]‖p[3]‖…‖p[L]‖salt)$
……
$RP_L=H_L(p[1]‖p[2]‖…‖p[L-1]‖salt)$
。
【００３４】
第一段階では、ユーザーはコンピュータ１１０のユーザー・インターフェース１１３を使ってパスワード・エントリーP（以下ではエントリーと命名される）を入力する（S11）。コンピュータ１１０は次いでそのエントリーを、耐性関数を使って処理して（S12）、複数のサブエントリー[P]を得る。複数のサブエントリー[P]は次いでサーバー１２０に送られる。当業者は、サーバー１２０への転送の間に前記複数のサブエントリー[P]を保護するために、特許文献４に記載される安全な認証されたチャネル（SAC: Secure Authenticated Channel）のような認証を使うことが好ましいことを認識するであろうが、当業者によく知られており、本開示の範囲外であるので、これについては説明しない。当業者はまた、パスワード空間でのパスワード・エントリーPをSACを通じてサーバー１２０に転送し、次いでサーバーが処理段階S12を実行することが可能であることも認識するであろう。
【００３５】
サーバー１２０は複数のサブエントリー[P]を受領し、各サブエントリーを、当該ユーザーについての記憶されているパスワード検証子と比較して、それらが一致するかどうかを判定する（S13）。少なくとも一つのサブエントリーが、あるパスワード検証子に一致すれば、ユーザーは認証され、通知がコンピュータ１１０を介してユーザーに送られる（S14）。どのサブエントリーもパスワード検証子に一致しなければ、ユーザーは権限を与えられず（S15）（すなわち、認証されず）、好ましくはこのことを通知される。ユーザーは、エントリーを入力する新たな試行をオファーされうることが可能である。
【００３６】
図示される方法は、
【００３７】

【数１】

$$\binom{L}{k} = \frac{L!}{(L-k)!k!}$$

個の異なるパスワード検証子およびサブエントリーを生成することによって、k個までのタイプ誤りを扱うよう、容易に一般化されることができる。ここで、各パスワード検証子およびサブエントリーは、（パディングされた）基準パスワードからk個のキャラクタを省くことによって生成される。

【００３８】

kが増すと、システムの安全性が低下し、計算および記憶上の要求が増すことが理解されるであろう。たとえば、L＝20については、k＝1は20個の値に対応し、k＝2は190個の値に対応し、k＝3は1140個の値に対応する。

【００３９】

k＝1については、本方法は、下記のアルゴリズムとして記述できる。これは好ましくは、タイミング攻撃を防ぐために一定時間において実装される。

【００４０】

【表１】

アルゴリズム1：タイプミスに耐性のあるパスワード検証（高々1個のミスタイプされたキャラクタ）

入力：候補パスワード：$\hat{p}[1]||\ldots||\hat{p}[L]$；　ソルト値：salt
出力：status（すなわち、入力されたパスワードが受け容れられるならstatus＝1）
（メモリ内　：$H[1],\ldots,H[L]$）

```
1: status ← 0
2: for i = 1 to L do
3:     Ĥ ← H_i(p̂[1]|| ... ||p̂[L] \ p̂[i] || salt)
4:     if Ĥ = H[i] then
5:         status ← 1
6:     end if
7: end for
8: return status
```

アルゴリズム１はサブエントリーiがパスワード検証子iと一致するかどうかを検査するだけであることが見て取れるであろう。これは、あるサブエントリーと当該ユーザーについてのすべてのパスワード検証子との間の比較よりも高い安全性につながる。

【００４１】

k＝2については、本方法は、下記のアルゴリズムとして記述できる。これは好ましくは、タイミング攻撃を防ぐために一定時間において実装される。

【００４２】

【表２】

アルゴリズム2：タイプミスに耐性のあるパスワード検証（高々2個のミスタイプされたキャラクタ）

入力：候補パスワード：$\hat{p}[1]||\ldots||\hat{p}[L]$；　ソルト値：salt
出力：status（すなわち、入力されたパスワードが受け容れられるならstatus＝1）
（　メモリ内　：$H[1,2],\ldots,H[1,L],\ldots,H[2,3],\ldots,H[2,L],\ldots,H[L-1,L]$）

1: status $\leftarrow 0$
2: for $i = 1$ to $L-1$ do
3:　　for $j = i+1$ to L do
4:　　$\hat{H} \leftarrow H_{i,j}(\hat{p}[1]||\ldots||\hat{p}[L] \setminus \hat{p}[i],\hat{p}[j]\ ||\ \text{salt})$
5:　　if $\hat{H} = H[i,j]$ then
6:　　status $\leftarrow 1$
7:　　end if
8:　　end for
7: end for
8: return status

メモリに記憶されるパスワード検証子のテーブルの肥大を緩和するため、計算に対して記憶をトレードすることが可能である。たとえば、2個のミスタイプされたキャラクタをサポートするために

【００４３】

【数２】

$$H[i,j] = H_{i,j}(p[1]||\ldots||p[L] \setminus p[i],p[j]\ ||\ \text{salt})$$

を使う代わりに、1個のミスタイプされたキャラクタについてのL個のパスワード検証子から出発して、検証アルゴリズムをしかるべく修正することが可能である。これは下記のアルゴリズム３に示されている。例解の目的のため、キャラクタは1バイト（0から255までの値の範囲を取る）で符号化されることが想定されている；アルゴリズムはL個の記憶されているパスワード検証子および256L(L－1)回のハッシュ関数評価を必要とするが、これはアルゴリズム２におけるL(L－1)/2個の記憶されているパスワード検証子およびL(L－1)/2回のハッシュ関数評価と比較される。

【００４４】

【表３】

アルゴリズム3：タイプミスに耐性のある、メモリ効率のよいパスワード検証
（高々2個のミスタイプされたキャラクタ）

入力：候補パスワード：$\hat{p}[1]||\ldots||\hat{p}[L]$；ソルト値：salt
出力：status（すなわち、入力されたパスワードが受け容れられるならstatus＝1）
（メモリ内：$H[1],\ldots,H[L]$）

```
1: status ← 0
2: for i = 1 to L do
3:     for j = 1 to L do
4:         if j ≠ i
5:             bak ← p̂[j]
6:             for Char = 0 to 255 do
7:                 p̂[j] ← Char
8:                 if H_l(p̂[1]|| ... ||p̂[L] \ p̂[i] || salt) = H[i] then
9:                     status ← 1
10:                end if
11:            end for
12:            p̂[j] ← bak
13:        end if
14:    end for
15: end for
16 return status
```

上記のアルゴリズムと同様に、アルゴリズム３は好ましくは、タイミング攻撃を防ぐよう、一定時間で実装される。

【００４５】

上記の諸アルゴリズムにおいて、種々の関数Hがそれらのアルゴリズムを通じて同じであり、有利にはSHA-3であることが好ましい。

【実施例】

【００４６】

基準パスワードがTECHNICOLORであり、1個までの誤りを受け容れることができるパスワード検証子がECHNICOLOR*、TCHNICOLOR*、TEHNICOLOR*、TECNICOLOR*、TECHICOLOR*、TECHNCOLOR*、TECHNIOLOR*、TECHNICLOR*、TECHNICOOR*、TECHNICOLR*、TECHNICOLO*、TECHNICOLOR,であるとすると、下記のミスタイプされたパスワード・エントリー（キャラクタが欠けているところのほかは誤りはボールドで示されている）は、下記の表に示されるように受け容れられるまたは拒否される。

【００４７】

【表４】

エントリー	一致するサブパスワード	受け容れ
TECBNICOLOR	TECNICOLOR*	Yes
TECHNICOLO	TECHNICOLO*	Yes
TECHNICOLORE	TECHNICOLOR	Yes
TECHHJCOLOR	-	No

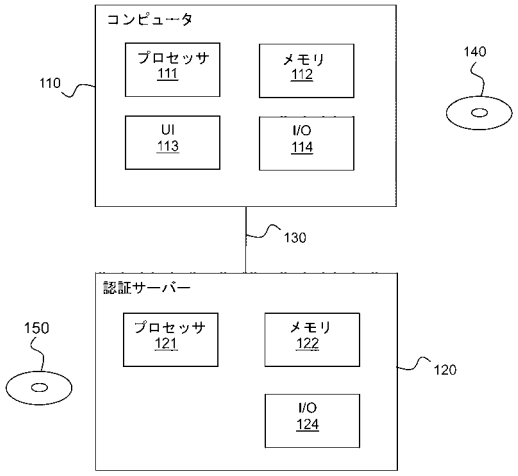
本開示が、タイプ誤りに耐性があり、比較が保護された空間においてなされることができるパスワード・システムを提供できることが理解されるであろう。

【００４８】

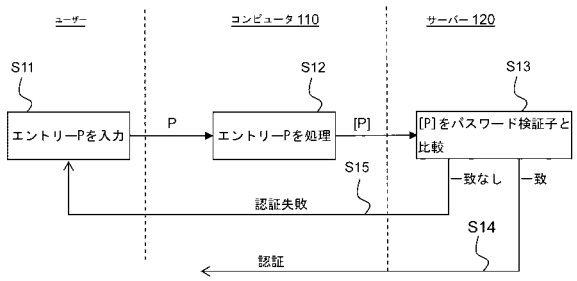
本記述および（適切な場合には）請求項および図面において開示される各事項は、独立して、または任意の適切な組み合わせにおいて提供されてもよい。ハードウェアで実装さ

れるとして記述される事項がソフトウェアで実装されてもよく、逆にソフトウェアで実装されるとして記述される事項がハードウェアで実装されてもよい。請求項に現われる参照符号は単に例解のためであり、請求項の範囲に対していかなる限定する効果ももつものではない。

【図１】



【図２】



フロントページの続き

(72)発明者　マルク　ジョイエ
フランス国　３５５７６　セソン・セヴィニェ　シー・エス　１７６　１６　ザック・ド・シャン・ブラン　アヴェニュ・ド・シャン・ブラン　９７５　テクニカラー・アールアンドディー・フランス

【外国語明細書】
2016004580000001.pdf