Vol.2019-GN-108 No.5
Vol.2019-SPT-33 No.5
2019/5/9

パスワード推測攻撃耐性を持つブレインウォレット

川上 智樹[1]　金岡 晃[1,a)]

概要：インターネット上の仮想通貨であるビットコインを管理するための仕組みとして、ビットコインウォレットが存在する。ビットコインウォレットは、署名鍵の保管方法の違いによって複数の形式に分類され、各形式でセキュリティの高さやユーザビリティの高さが異なる。ビットコインウォレットの一種であるブレインウォレットは、パスワードの入力のみでビットコインのやり取りができるためユーザビリティが高く、署名鍵の情報を端末に保管しない。そのためマルウェア感染などによって署名鍵が漏洩する恐れもない。その反面、総当たりで候補を試すパスワード推測攻撃には耐性がなく、使用に際してセキュリティ面での問題を抱えている。本研究では、ブレインウォレットの持つ高いユーザビリティを損なうことなく、推測攻撃に対して耐性を持たせるためのアプローチを 2 種類提案する。そのうち、ブレインウォレットに前処理を挿入するアプローチについて被験者実験を行い、提案したアプローチがユーザビリティに与える影響を調査した。

Brain wallet resistant to password guessing attacks

Tomoki Kawakami[1]　Akira Kanaoka[1,a)]

1. はじめに

インターネット上の仮想通貨であるビットコインの利用者は年々増加傾向にあり, 2012 年 9 月に約 2 万件だったウォレット数は, 2019 年 1 月 29 日時点で約 3,300 万件と, 6 年間で 1,000 倍以上も増加している [1]。ビットコインは改竄が困難な点や, 特定の管理者を必要としない非中央集権的な決算方法であることが注目される一方で, 取引の増加によるスケーラビリティの問題や, ビットコインのもつ匿名性を悪用してマネーロンダリングに利用されるなど, さまざまな問題点も存在する。また, 実用面においてもビットコインの送金に必要とされる秘密鍵の管理などの問題が挙げられる。

ビットコインの秘密鍵及びビットコインアドレスの管理のためにビットコインウォレットと呼ばれる仕組みが存在する。ビットコインウォレットは, 秘密鍵の管理方法の違いによって複数の形式に分類され, それぞれセキュリティ強度の高さやユーザビリティの高さなどが異なる。

[1] 東邦大学
Toho University, Chiba, 274–8510, Japan
[a)] akira.kanaoka@is.sci.toho-u.ac.jp

本研究では, ビットコインウォレットの一種であるブレインウォレットに着目する。ブレインウォレットは秘密鍵の情報をデバイスに格納せず, 秘密鍵のもととなる公開鍵ペアを生成するためのパスワードをユーザ自身が記憶する方式である。Eskandari らの論文 [3] によると, ブレインウォレットは他の形式のビットコインウォレットと比較して, ユーザビリティが高いとされている。秘密鍵をデバイスに格納する必要がないため, マルウェアによって秘密鍵が盗難される恐れがない点もブレインウォレットの利点である。しかし, Vasek らの論文 [4] では, ブレインウォレットは総当たり的なパスワード推測に対しての耐性がなく, パスワード推測攻撃によってユーザの秘密鍵が特定される危険性があることが指摘された。

本研究では, ブレインウォレットの持つ高いユーザビリティを損なうことなく, Vasek らが行ったパスワード推測攻撃に対しての耐性を与えることで, 高いユーザビリティをもちつつ, 実用に足るセキュリティ強度を兼ね備えたビットコインウォレットを提案する。

ブレインウォレットに推測攻撃耐性をもたせるためのアプローチとして, ブレインウォレットの鍵生成時に機密情

報 α を加えるアプローチ, ブレインウォレットに計算量を増やす前処理を挿入するアプローチの 2 種類を提案した。ブレインウォレットに前処理を挿入するアプローチについては, 実際に提案アプローチを採用したアプリケーションを作成し, パフォーマンスの評価及びユーザビリティの評価を行うことで, 提案アプローチが従来のブレインウォレットに与える影響を調査した。調査の結果として, 本研究で提案したブレインウォレットにストレッチング処理を挿入するアプローチによるユーザビリティの変化は確認されず, 従来のブレインウォレットのもつ高いユーザビリティを損なうことなく, 改良を加えることに成功した。

2. 関連研究

ブレインウォレットでは, 秘密鍵及びビットコインアドレスが一つのパスワードから決定論的に算出される。同じパスワードからは毎回同じ秘密鍵とビットコインアドレスが生成される。

そのため, 公開台帳に記録されたビットコインアドレスと, あるパスワードから生成したビットコインアドレスとが一致する場合, 入力したパスワードは同じであると推測できる。これはブレインウォレットに対してパスワードの推測が可能ということを意味する。また, ビットコインのすべての取引履歴は公開台帳に記録されているため, 公開台帳を参照することで今まで取引に使用されたすべてのビットコインアドレスを誰でも確認することができる。

Vasek らの論文 [4] では, 過去に流出した大規模なパスワードセットなどの利用可能なソースから単語やフレーズのリストを収集し, 約 3,000 億のパスワードセットを構築した。そして公開台帳に記録されたすべてのビットコインアドレスと, 候補パスワードセットから生成したビットコインドレスとを比較した結果, 公開台帳に記録されたビットコインアドレスと, 候補パスワードから生成したビットコインアドレスが一致する組み合わせを 844 件分特定した。

3. 提案手法

3.1 解決すべき課題

ブレインウォレットを改良するにあたっては, 以下の 3 つの要件を満たす必要がある。

(1) パスワード推測攻撃への耐性：ブレインウォレットは, パスワード推測攻撃によって秘密鍵が特定される恐れがある。そのため, 推測攻撃に対して耐性を持たせる必要がある

(2) 内部構造の公開：ブレインウォレットの性質上, 内部構造を利用者に対してオープンにする必要がある。そのため, 攻撃者が内部の構造を把握しているという前提で, 実用に足るセキュリティ強度を実現しなければならない

(3) ユーザビリティの確保：ブレインウォレットのもつ長所を活かすためには, 従来のブレインウォレットがもつユーザビリティの高さを損なうことなく, 実用に足るセキュリティ強度を実現する必要がある

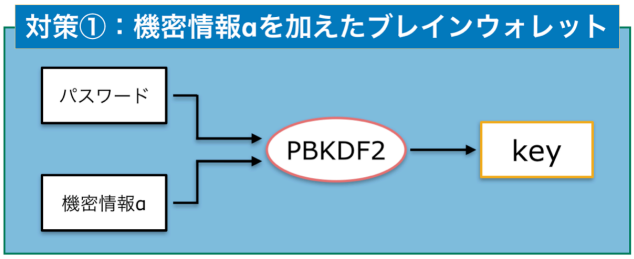

図 1 ブレインウォレットに機密情報 α を加え, 推測攻撃を防ぐアプローチ

3.2 解決に向けたアプローチ

3.2.1 アプローチ 1. 機密情報 α を加える

この対策は, ブレインウォレットによるビットコインアドレス及び秘密鍵の生成過程において, パスワード以外の「機密情報 α」を加えるというアプローチである。攻撃者は機密情報 α が特定できないかぎり, パスワード推測攻撃による推測を成立させることができなくなる。

機密情報 α の条件は, 4.1 の要件を踏まえると以下のようになる。

- ユーザが α を別途に記憶する必要がない
  - ユーザが別途に記憶する必要のある情報の場合, 記憶しなければならない単語数が増えることでユーザビリティに影響が出るだけではなく, 推測攻撃への耐性も本質的には上がらない。
- デバイスに α を格納する必要がない
  - 機密情報 α をデバイスに格納してしまう場合, マルウェア感染によって機密情報 α が漏洩するおそれがでる。秘密鍵の情報をデバイスに保管するハードウェアウォレットと実質的な変わりがない仕組みとなってしまう。
- ウォレットをマルチデバイスで利用可能
  - ブレインウォレットの利点の一つとして, デバイスに依存しないということが挙げられる。そのため, PC やスマートフォンといったデバイスに関係なくウォレットが使用可能である必要がある。

3.2.2 アプローチ 2. 前処理を挟み計算量を増やす

この対策は, ブレインウォレットによるビットコインアドレス及び秘密鍵の生成過程に置いて, 計算量を増やす前処理を挟むことで, パスワード推測攻撃に必要な計算量を大幅に増やすというアプローチである。

前処理の条件は, 4.1 の要件を踏まえると以下のようになる。

- ユーザサイドは負担が軽い
  - 処理にかかる時間がユーザにとって許容できる範囲

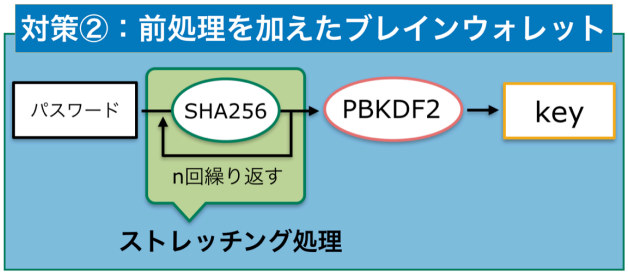

図 2　ブレインウォレットに前処理を挟み, 攻撃者の計算量を大幅に増やすアプローチ

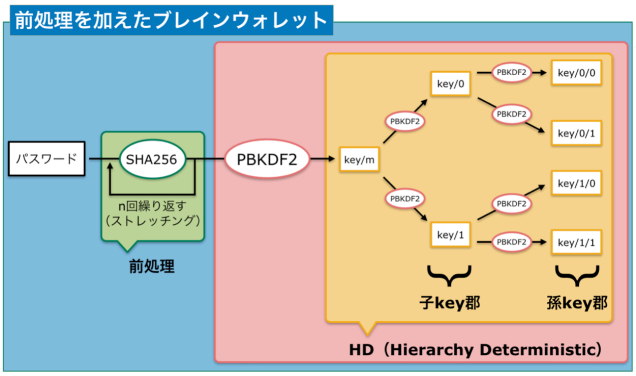

図 3　ストレッチング処理を挿入したブレインウォレット

内である。

- 攻撃者サイドは負担が重い
  - パスワード推測にかかる時間が攻撃者にとって許容範囲外である。

3.3　具体的構成

3.3.1　ストレッチング処理

ストレッチング処理とは, ハッシュ値の計算を数千回～数万回繰り返し行うことである。攻撃にかかる計算時間を長くできるため, 総当たりでパスワードを推測するような攻撃手法への対抗策として有効である。

3.3.2　ストレッチング処理を挿入したブレインウォレット

今回提案するアプローチでは, パスワードを鍵導出関数に入力する前にストレッチング処理を行い, 従来のモデルよりも計算量を大幅に増やす。代表的な鍵導出関数である PBKDF2 ではそれ自身にストレッチング処理を内包しており、その回数も設定可能であるが、本研究ではストレッチング処理を独立化させることにより鍵導出関数の選択へ自由度を与えた。また, ウォレットの内部構造には BIP-32 にて提案された HD ウォレットを用いることで, 一つのパスワードで複数のビットコインアドレス及び秘密鍵を管理する。

4.　ユーザビリティ調査のための実験方法の詳細

4.1　実験

4.1.1　実験の目的

本研究で提案したアプローチを採用したブレインウォレットと, 従来のブレインウォレットを被験者実験を通じて比較し, 提案したアプローチがユーザビリティに与える影響を調査する。

4.1.2　評価方法

実験の評価方法として, 本研究で作成したアプリケーションを被験者に利用してもらい, System Usability Scale (SUS) のアンケートに回答してもらう。アプリケーションは 2 種類用意した。いずれもウォレット機能を持ち、ビットコインの疑似送信機能を持つアプリケーションであるが、1 つは提案手法のストレッチング処理を導入し（アプリ A)、もう 1 つはストレッチング処理を省いた既存ブレインウォレットと同様とした（アプリ B)。アンケート結果の集計によって得られた点数差により, アプリ A とアプリ B とのユーザビリティの差の測定を行う。この実験において使用する SUS は, 原文の 10 項目を和訳したものを使用した。

今回の実験では SUS の項目における「システム」という表記を「アプリケーション」に変更してアンケートを行った。被験者にとって「システム」が何を指し示しているかが不明確になるおそれを考慮したためである。また, 被験者にアプリケーションのユーザビリティを調査していることをより意識させるためにも, アプリケーションと表記した。

4.1.3　実験の方法

今回の実験では, 被験者の意識や振る舞いをより深く調査するため, アンケートの回答に対しての質問を行った。アンケートの各項目に対してなぜその数値を選んだのかと質問を行うことにより, 被験者がアプリケーションに対してどのように感じたのかをより深く観測することが可能であると考えた。また, ユーザ行動に偏りが生じてしまうことを避けるため, 本来の実験目的とは異なる仮の実験目的を立て, 被験者の意識をアプリケーションから逸らすようにした。

(1) 仮の実験目的の説明
　初めにブレインウォレットの使いやすさから被験者の意識を逸らすため, 仮の実験目的を被験者に説明する。

(2) 実験内容の説明を行い, 被験者にアプリケーションを利用してもらう
　被験者にアプリケーションを利用してもらい, 使いやすさを確かめてもらう。実験の内容は以下の通りである。
　被験者は以下の作業を 1 セットとし, 計 10 セット

行う。
(a) 時間計測（ストップウォッチ）開始
(b) 宛先ユーザの選択
(c) 送金額の入力
(d) 使用ウォレットの選択
(e) パスワードの入力
(f) 送金
(g) ストップウォッチ停止

(3) 本来の実験目的の説明

(4) 本来の実験目的で実験結果を利用することに対しての同意を得る
被験者に本来の実験目的をあらためて伝え, 実験の結果を本研究の分析に利用することの同意を得る。同意が得られない場合は録画した映像など実験中に記録したものを破棄し, 実験の報酬を渡す。

(5) SUS アンケートに回答してもらう

(6) アンケートと実験に対する質問
アンケートの質問は, 各項目に対してなぜその数値を選んだのかの理由を聞く。すべての被験者に対して 4 つの質問と, 実験中に特徴的な動きをした被験者に対しては, その行動の意図を問うための質問を行う。すべての被験者に対して行う 4 つの質問は, 以下の通りである。
(a) 今回行った実験内容・目的について聞く以前に, ビットコインウォレットを利用したことがあるか
質問の意図：ビットコイン及びビットコインウォレットの認知についての質問
(b) 効率よくビットコインウォレットが利用できたと感じたか
質問の意図：ユーザビリティの要素である「効率性」「有効性」についての質問
(c) アプリケーションを操作している際にストレスを感じたか。また, ストレスを感じた場合はその要因をどこであると思うか
質問の意図：ユーザビリティの要素である「満足度」についての質問
(d) アプリケーションを操作している際の実行時間についてどう感じたか
質問の意図：ユーザビリティに関係すると思われる「実行時間」についての質問
すべての被験者に対して行う 4 つの質問にはそれぞれ意図が存在し, アプリケーションのユーザビリティの要素についての満足度や効率性の認知を測るものとなっている。

(7) 被験者から, 実験についての質問・感想をもらう

4.1.4 被験者について

被験者は情報科学を学ぶ学生であり, 且つ本研究についての前提知識を持たない学生から募集した。また, 被験者の募集方法は, 仮の実験目的・所要時間・実験の対象者・実験報酬等を記載したチラシを学内で配布することとした。

被験者実験に対しての報酬は, いくつかの被験者実験を行っている論文を見る限り, 米国の論文では報酬が 1 時間当たり 10 ドル程度であったが [6], 最低賃金が 15 ドルへ引き上げられた [7] あとから 15〜20 ドルへと変化しており [8][9][10], その国の 1 時間当たりの最低賃金を基準とした報酬額であることが見て取れた。そのため, 千葉県の最低賃金が時間給 895 円であり [11], 実験予定時間が 30 分であることを踏まえ, 最低賃金の時間給の半分の金額にあたる 500 円が妥当であると考え, 500 円分の図書カードを報酬とした。

図 4 実験風景 1

図 5 実験風景 2

4.1.5 実験風景

以下に実験中の風景を記載する。

4.2 アプリの実装と利用端末

4.2.1 実装内容

アプリは Apple iOS アプリケーションとして実装し、実

図 6　アプリケーションのメイン画面

験時の扱いやすさ等を考え iPad で動作させること意識した作りとした。HD ウォレットの実装には, GitHub 上で公開されている HDWallet ライブラリ [5] を利用した。

鍵導出関数には PBKDF2 を用い、ストレッチング処理に用いるハッシュ関数には SHA-256 を採用した。

4.2.2　アプリ画面

4.2.3　利用端末

本研究における実装とその後の評価には、9.7 インチ iPad（第 6 世代）, OS バージョン：12.1, CPU：2.34GHz, プロセッサ数：2 を用いた。

5.　提案手法の評価

5.1　パフォーマンスの評価

本研究では, パフォーマンス評価用のアプリケーションを作成し, HD ウォレットの有無やストレッチング処理の有無によるパフォーマンスへの影響を調査した。パフォーマンスの評価実験では, HD ウォレットの階層数を 5 層に統一し, ストレッチング処理の回数と鍵生成に要する実行時間について評価した。

実験の結果, 表 1 のようなことがわかった。実行時間は, それぞれ 100 回の試行結果を平均した値である。また, 表中のストレッチング 0 回は, 提案アプローチを非採用の, 既存のブレインウォレットを示すものであり, 実行時間は HD ウォレットに寄る鍵生成の時間だけが示される。並びに, HD ウォレットの階層数が 5 層での鍵生成は非常に短時間ででき, 100 回の試行を平均した結果では, 測定環境では有効な数字が得られなかった。

ストレッチング回数と実行時間は概ね正の比例関係にあった。以上の結果を踏まえ, 30,000 回程度であればユーザが許容できる範囲内だと推測し, ユーザビリティ評価のための被験者実験では, ストレッチング処理の回数を 30,000 回とした。

ストレッチング回数	実行時間
0 回	0.00 秒
10,000 回	2.28 秒
20,000 回	4.33 秒
30,000 回	6.08 秒
50,000 回	9.57 秒
100,000 回	18.41 秒

表 1　ストレッチング回数と実行時間

表 2　実験被験者の情報と使用したアプリケーション

ID	性別	学年	実験で使用したアプリ	SUS スコア
1	男性	2	ストレッチングあり	80.0
2	男性	3	ストレッチングあり	85.0
3	男性	3	ストレッチングあり	85.0
4	男性	2	ストレッチングあり	60.0
5	男性	2	ストレッチングなし	77.5
6	男性	3	ストレッチングなし	95.0
7	男性	4	ストレッチングなし	70.0

5.2　ユーザビリティの評価

5.2.1　System Usability Scale（SUS）

今回の実験において, 集まった被験者のデータは 7 名分であった。被験者人数が少ないことが見受けられるが, Nielsen らの論文 [12] によれば, ユーザテストは 5 人ほどで実施した場合であっても, ユーザビリティに関する有益なデータが得られることが示されている。

被験者の情報と算出した SUS のスコアを以下の表 2 にまとめた。

SUS のスコアからは, ストレッチング処理の有無によるユーザビリティの変化は見られなかった。

5.2.2 Grounded Theory Approach（GTA）

実験中の行動および事前知識についてインタビューを行い, Grounded Theory Approach (GTA) を用いて質的分析を行った。GTA を行った結果，以下の 3 個の概念 （カテゴリ）に分類した.

5.2.2.1 シンプル・簡単

提案アプローチの採用・非採用にかかわらず, シンプルであることや簡単であることをコメントした被験者が多かった。そのコメントについても, SUS の質問項目を問わず, 広くコメントがあった。SUS の 1, 2, 3, 4, 7, 8, 9, 10 番と多岐にわたる。そのため, ブレインウォレットの単純機能アプリケーションとしてのシンプルさは明確になったと考えられる。

シンプルさや簡単さについて言及したコメントは, ストレッチング処理あり：17, ストレッチング処理なし：11 と, 提案アプローチの採用・非採用による影響も見て取れなかった。

5.2.2.2 使用経験

今回の被験者全員が, これまでにビットコインウォレットの利用経験がないとしていた。それに従い, SUS の質問項目での機能統合についてや, 利用前の事前知識の必要性についての回答において「使用経験がないため判断が難しい」といったコメントがあった。また, 被験者の 1 人から

「ウォレットの前提知識が必要（ID：5)」

とのコメントがあったが, その他の被験者からは前提知識の必要性について言及したコメントは見られなかった。

5.2.2.3 機能への要求

実験アプリケーションへのコメントで, 本研究が評価対象としたいストレッチング処理の有無によるユーザビリティ変化以外の部分で, アプリケーションが持つ機能へのコメントがいくつかあった。

「送金先やウォレットの項目をそれぞれ選択するよりも, 直線的に次の入力候補が表示されるほうがいい（ID：8)」

金銭的なリスクを認識し, 機能のシンプルさや簡単さなどを逆に不安視するコメントもあった。

「簡単すぎて逆に不安かもしれない（ID：5)」, 「特に使いづらい点はないが, 使いやすすぎて不安だった（ID：3)」

被験者 ID：3 は別の項目でのコメントにおいても, 不安視している点について言及している。

「送金という責任の伴う行為に対して, 捜査が簡単すぎると思った（ID：3)」

などは, 特徴的なコメントであろう。

それに付随して, 機能としてのチェック機能の必要性を挙げる被験者もいた。

「チェック機能がほしいかもしれない（ID：3)」

シンプルさや簡単という回答が多数を占める一方で, なかには本アプリケーションの面倒な点を挙げる被験者もいた。

「小数を打つのが面倒だった（ID：2)」, 「入力のし直しが面倒（ID：4)」

不安視する点をコメントした被験者は, 提案アプローチを採用のアプリケーションで実験した被験者が 1 人, 非採用のアプリケーションで実験した被験者が 1 人だった。

ストレッチング処理による影響を, 一番大きく受けると考えられる実行時間について言及するコメントも見られた。

「ストレスなく, 実行も早いと思った（ID：1)」, 「スムーズだった（ID：6)」

これらのことから, ストレッチング処理の有無による違いは被験者グループ間で大きな差はなく, ストレッチング処理により被験者にストレスを与えたなどの影響は, ここからは見られないと考えられる。

6. 今後の課題

本研究は, ブレインウォレットにストレッチング処理を挿入することで総当たり的な推測攻撃に耐性をもたせるというアプローチを提案し, 被験者実験を通して提案アプローチがユーザビリティに与える影響を調査した。調査の結果として, 提案アプローチがユーザビリティに与える影響が少ないことが示され, ブレインウォレットのもつユーザビリティの高さを損なうことなく改良を加えるとこに成功した。その一方で, 本研究においては提案アプローチを採用したブレインウォレットの推測攻撃耐性についての実験を行うことができなかった。そのため, 今後の課題として提案アプローチを採用したブレインウォレットに対して実際に攻撃を仕掛け, セキュリティ面での強度を測定する必要がある。そうすることで, 提案アプローチを採用したブレインウォレットがユーザビリティ及びセキュリティの両面で, 実用に足る水準に達しているかどうかの判断が可能になる。

7. まとめ

本研究は, ユーザビリティが高く実用に足るセキュリティ強度を持ったビットコインウォレットを提案すべく, ユーザビリティは高いが推測攻撃に対して耐性を持たないブレインウォレットに着目した。ブレインウォレットの持つ高いユーザビリティを損なうことなく, 推測攻撃に対して耐性をもたせるためのアプローチを 2 種類提案し, ストレッチング処理を挿入するアプローチについては, 提案アプローチを採用したアプリケーションを作成し, パフォーマンス評価及びユーザビリティ評価を行った。

パフォーマンス評価の方法としては, ストレッチング処理の回数とそれに要する実行時間の関係を調査し, ストレッチング処理の回数と実行時間は概ね正の比例関係にあることを観測した。

ユーザビリティ評価の方法としては, 被験者実験を行う

ことで提案アプローチがユーザビリティに与える影響を調査した。提案手法のユーザビリティを評価するために, 量的分析方法として System Usability Scale（SUS）を使用し, 質的分析手法として Grounded Theory Approach（GTA）を使用してユーザビリティ評価を行った。提案したアプローチを採用したブレインウォレットを利用した被験者からは「ストレスなく, 実行も早いと思った」といった, ユーザビリティに対する好意的な回答も確認できた。その一方で, 本研究においては提案アプローチを採用したブレインウォレットの推測攻撃耐性についての実験を行うことができなかったため, その点は今後の課題としたい。

参考文献

[1] Blockchain Wallet Users, https://www.blockchain.com/ja/charts/my-wallet-n-users?timespan=all
[2] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", 2008
[3] Shayan Eskandari, David Barrera, Elizabeth Stobert, and Jeremy Clark, Concordia University, ETH Zurich, Carleton University, "A First Look at the Usability of Bitcoin Key Management", USEC 2015, 2015
[4] Marie Vasek, Joseph Bonneau, Ryan Castellucci, Cameron Keith, and Tyler Moore, "The Bitcoin Brain Drain: Examining the Use and Abuse of Bitcoin Brain Wallets", 20th International Conference on Financial Cryptography and Data Security, FC 2016
[5] HDWallet, https://github.com/essentiaone/HDWallet
[6] S. Ruoti, N. Kim, B. Burgon, T. van der Horst, K. Seamons. Confused Johnny： When Automatic Encryption Leads to Confusion and Mistakes（SOUPS 2013）,2013
[7] 米最低賃金アップ, http://jp.wsj.com/articles/SB12692037482832534161104582049733755391666
[8] S. Ruoti, J. Andersen, A. Heidbrink, M. O'Neill, E. Vaziripour, J. Wu, D. Zappala, K Seamons. "We're on the Same Page"：A Usability Study ofSecure Email Using Pairs of Novice Users（ACM CHI2016）, 2016
[9] S. Ruotiy, J. Anderseny, T. Hendershoty, D. Zappalay, K. Seamons. Private Webmail 2.0: Simple and Easy-to-Use Secure Email（UIST2016）,2016
[10] W. Bai, D. Kim, M. Namara, Y. Qian, University of Maryland, College Park; P. G. Kelley, University of New Mexico; M. L. Mazurek, University of Maryland, College Park（SOUPS 2016）,2016
[11] 千葉県最低賃金, https://www.city.ichihara.chiba.jp/kanko/0205sangyou/hatarakuhito/info/saiteichingin.html
[12] Jakob Nielsen, Thomas K. Landauer, "A mathematical model of the finding of usability problems.", In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93). ACM, New York, NY, USA, 206-213.