学術情報処理研究 No.20
2016年9月

ユーザ負担を考慮したワンタイムパスワード認証システム

One-time password authentication system with small burden for users

多田 充 *
Mitsuru Tada
千葉大学 統合情報センター
Institute of Media and Information Technologies, Chiba University
〒263-8522 千葉市稲毛区弥生町 1-33
Yayoicho 1-33, Inage, Chiba 263-8522, Japan

概要

ワンタイムパスワード認証は，固定パスワード認証で問題視されている安全性を解決するものとして，金融関係サイトだけではなく，大学組織のサービスシステムに対しても，近年重要視され実際普及してきている。本論文の著者は，(固定) パスワードで制御されているサービスシステムにおける認証の安全性を強化する方法について，2015 年の学術情報処理研究集会で発表した [9]。本論文では，このシステムを拡張することにより，ワンタイムパスワード認証システムを構築する。ワンタイムパスワードは単独で利用されることは少なく，多くのサービスシステムでは固定パスワードと組み合わせて利用される。近年，多くの大学組織において，組織内の各サービスシステムで利用される (固定) パスワードは共通なものになっていることが多くなり，複雑なパスワードをいくつも覚えるというユーザの負担は軽減された。しかし，ユーザ ID については，システム毎に異なった文字列を用いていることがあり，その種類数が多くなるとその記憶負担も苦情の種となりうる。本論文で述べるワンタイムパスワード認証システムでは，ユーザが覚える文字列が (固定) パスワード 1 つのみとなるので，ユーザの負担をより軽減させることが可能となる。

キーワード：ワンタイムパスワード認証，3 者間認証，2 要素認証

* E-mail: m.tada@faculty.chiba-u.jp

1 はじめに

インターネット上のサービスシステムでユーザ認証を行うものの大半は，「ユーザ ID および固定パスワード」によるものである。IPA による調査 [1] では，その調査対象となったサービスシステムの全てで固定パスワードが用いられている。固定パスワードによるユーザ認証の安全性に関する問題点は，既に多くの専門家により指摘されていることであり，実際，総当たり (ブルートフォース) 攻撃，辞書攻撃，いわゆるリバースブルートフォース攻撃，リスト型アカウントハッキングなど，さまざまな攻撃手法により，脅威に晒されている。

一方，利便性という面で見てみると，「ユーザ ID &固定パスワード」による認証方式は，サービスシステムの運用側にとって，導入しやすい・コストが安いなどの利点があり，ユーザにとっても解りやすい・馴染みやすいものであるが，利用するサービスシステムの個数が多いユーザにとっては，

(1) それぞれのサービスシステムに対して異なるパスワードを設定し，それらを全て覚えておかなければならない。
(2) それぞれのサービスシステムで用いるユーザ ID が異なることがあり，これらについても，ユーザ ID として何が (メールアドレス，利用者番号，自分が任意に定めた文字列，などのうちのどれが) 設定されているか，全て覚えておかなければならない。

などの問題がある。[1] によれば，53% 以上が 5 個以上のユーザ ID &パスワード (PW) を持っており，3 個以上だと 81% 以上にもなる。一方，記憶できる ID & PW の個数については，67% は 5 個の ID & PW を覚える自信がなく，32% は 3 個覚える自信がないようである。自身が覚えられる ID & PW の個数以上のサービスシステムを利用する場合は，手帳や紙などに書き留めておくか，パスワードマネージャーなどのソフトウェアを使用するしかない。そうでなければ，覚えられる程度の簡単な PW を設定するか，PW を使い回すことになり，他者によるパスワードクラッキングの危険性を高めてしまう。そのため，ネットバンキング等の金融サイトではワンタイムパスワード (OTP) 認証を採用するものが多くなってきている。

これは大学組織内ネットワーク上のサービスシステムでも同様で，特に組織外ネットワークからのアクセスに対して，認証レベルを上げるようになってきている。ワンタイムパスワード認証を採用している大学は [3, 4, 5, 7, 8] などで見受けられ，今後導入を検討している大学も多いと思われる。

ワンタイムパスワード認証は，導入コストが高くなるという問題があるが，それをクリアしたとしてもなお，構成員 (ユーザ) にあまり馴染みがなかったり，通常の (固定) パスワード認証に比べて不便，使い勝手が悪く，苦情の種になりうるという問題がある。とはいえ，昨今のセキュリティ事情から，いつまでも (固定) パスワードによる認証のみを採用し続けることも難しいので，ワンタイムパスワード認証方式の利便性を上げることは急務であると思われる。しかし，ワンタイムパスワード認証手法の多くは，(固定) パスワード認証のような「記憶情報による認証 (SYK)」だけではなく，ユーザの「所有物による認証 (SYH)」も行っており，つまり，2 要素認証を実施している[*1]。認証要素が増えることは，(SYK が増えた場合は) 覚えることが増える，(SYH が増えた場合は) 所持していなければならないものが増えるなど，ユーザの負担が増えることを意味する。例えば，ハードトークンを用いたワンタイムパスワード認証方式は比較的普及しているものであるが，ユーザはログイン時にそのトークンを持っていなければならないわけであり，それが煩わしさを与えることは否定できないであろう。

本論文では複数のサービスシステムに適用できるワンタイムパスワード認証システムでありながら，ユーザの負担を極力減らすことを目的としたシステムの構築方法を提案システムとして述べる。本システムにおいては，ワンタイムパスワードを取得する媒体は「ユーザ所有のスマートデバイス[*2]」であり，それを所有物認証に使用する，つまり，現在多くの人々が普段から持ち歩いていると思われる機器を使用するため，乱数表方式やハードトークン方式のような (新たに持ち物が増えるという) 煩わしさは生じない。普段から持ち歩いている機器を使用するということは，その紛失に気づきやすいという利点もある。また，ユーザのスマートデバイスとセンターとの 2 要素 (SYK, SYH) 認証をパスすることにより，ユーザ ID がサービスシステムから (センターを経由して) 通知されるようシステムを構築するため，複数のサービスシステムが，それぞれ異なった文字列をユーザ ID として採用していたとしても，ユーザはそれらを記憶する必要はない。つまり，サービスシステムの数に関係なく，ユーザが覚えるべき情報は

- センターと共有する (固定) パスワード 1 つのみ

[*1] マトリクス認証方式は，記憶情報のみの認証なので，認証要素数は 1 である。

[*2] 本論文では「スマートデバイス」と記述するが，この「スマートデバイス」はネットワーク通信機能が必要である。そのため，現時点では「携帯電話機器」とするのが現実的かもしれない。

である。 安全性の面では以下のように考えられる。本提案システムは，一般に普及しているマトリクス方式のような SYK のみの認証ではなく，SYK および SYH の 2 要素認証になっている。ワンタイムパスワードを電子メール (特に，携帯キャリアメール) で通知する方法も 2 要素認証として挙げられることがあるが，主な携帯キャリア会社が Web メールを提供していることもあり，電子メールは インターネットメール/携帯メール 問わず「ID & PW」，つまり記憶情報のみで取得可能であると考えられ，携帯メールを採用していたとしても完全な形の 2 要素認証とは考えにくいと思われる。本提案システムにおいて，ワンタイムパスワードは「ユーザが登録に使用したスマートデバイス」のみが取得可能である。さらに，本提案システムは，ハード/ソフト トークンによるシステムとは異なり，サービスシステムはユーザがワンタイムパスワード発行をリクエストして取得したことを知ることができるので，そのリクエストがない限り，(当該ユーザの) アカウントをロックすることが可能である[*3]。つまり，ユーザが上記の 2 要素認証をパスしない限り，不正ログインは完全に防御可能となる。

本論文は以下の通り構成される。第 2 章では，提案システムと同様のシステム構成をしている [9] の概要を述べる。[9] は，(固定) パスワード認証システムの安全性を強化させるためのアカウントロック機能 (認証シャッター) について述べられているが，本論文の提案システムはそれを拡張してワンタイムパスワード認証システムにしたものである。第 3 章では，提案システムの構成，および，提案システムにおけるユーザ登録手続き，認証 (ログイン) 手続きのプロトコルを述べる。第 4 章では，提案システムの安全性および利便性について述べ，第 5 章で本論文をまとめる。

2 3 者間認証 [9]

ここでは，著者が 2015 年に [9] で発表したシステムの概略を述べる。本論文の提案システムは [9] と同様の構成であり，ユーザ，サービスシステムおよびセンターからなる。

[9] では，既に固定パスワードによる認証を採用して運用されている (複数の) サービスシステムに対して，それらの安全性を強化するためのアカウントロック機能 (認証シャッター) を与える方法が述べられており，ユーザ登録手続きは，各サービスシステムにユーザ情報が登録されている状態からスタートする。具体的には，図 1 で概略を示すように，以下のプロセスを経て登録手続きが実行される。

(1) サービスシステムはセンターにユーザ追加リクエストを送る。
(2) センターは登録チケットを発行し，サービスシステムに送る。
(3) サービスシステムは登録チケットを安全にユーザに渡す[*4]。
(4) ユーザは自身が所有するスマートデバイス (で動作する専用アプリ) に登録チケットを入力し，センターに登録リクエストを送る。
(5) センターはユーザに登録が完了した旨を通知する。

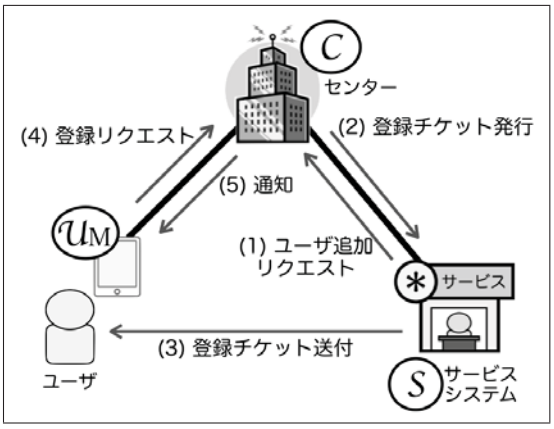

図 1　[9] における登録手続きのイメージ

ユーザ情報の流れという側面では，図 2 で示すように，元々，各サービスシステムが保有していたユーザ情報 (のリスト)uList の各アカウントに対して，それをサービスシステムも併せて紐付けできる識別子をセンターが管理することになる。第 3 章でも述べるが，提案システムにおいては，昨今のユーザ情報管理の形態を踏まえ，全てのユーザ情報はセンターにより (図 2 内では `aList` として) 一元的に管理される。

次に，[9] における認証手続きについて述べる。ユーザがサービスシステムを利用 (ログイン) する際，ユーザ，サービスシステムおよびセンター間で，3 者間認証を実行する。具体的には，図 3 のように，

(1) ユーザは，センターに対して，サービスシステムを利用開始できるようリクエストする。
(2) センターは，当該ユーザがログインできるよう，サービスシステムに指示する。
(3) サービスシステムは，(2) の指示に応答する。
(4) センターは，サービスシステムにログインできるようになったことをユーザに通知する。

[*3] [9, 10] ではこのロック機能を「認証シャッター」と呼んでいる。

[*4] 登録チケットの渡す手段は，確実にユーザに送ることができるのであれば オンライン/オフライン を問わない。

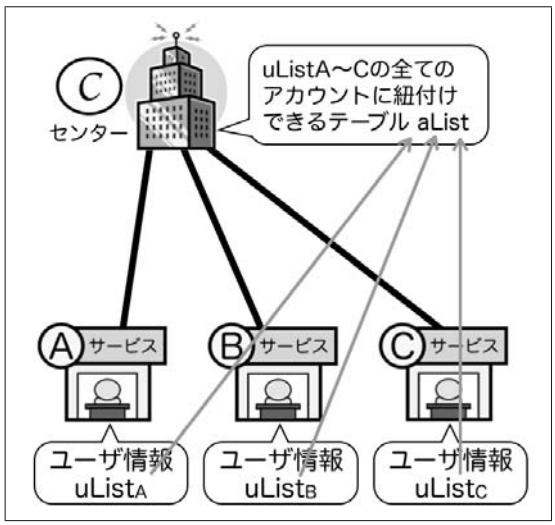

図 2　[9] におけるユーザ情報

(5) ユーザはサービスシステムにログインし利用を開始する。

という手続きを経る。具体的には，(1) の段階で記憶情

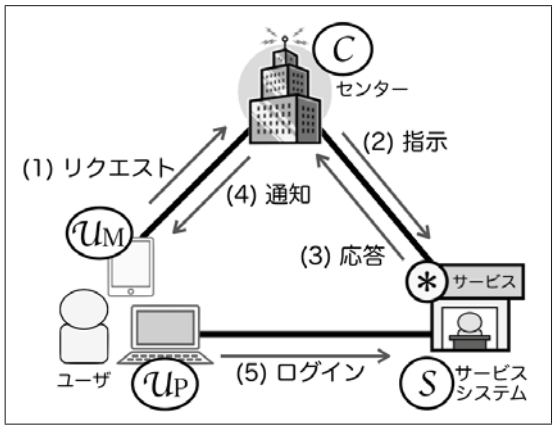

図 3　ログイン手続きのイメージ

報および所有物を用いた 2 要素認証を行い，(2) で当該ユーザのアカウントのロックを解除する。サービスシステムは (3) においてロック解除の期限を知らせ，(4) でセンターはその旨をユーザに通知する。ユーザは (5) において，予め定められたユーザ ID と固定パスワードを用いてサービスシステムにログインする。

3　提案システム

本章では提案システム構成およびプロトコルについて述べる。提案システムに登場するエンティティは，システムを利用するユーザ $\mathcal{U}$，ユーザがそのサービスを利用するサービスシステム $\mathcal{S}$，ユーザ情報を統合的に管理するセンター $\mathcal{C}$，および登録作業 (の一部) を行うシステム管理者 $\mathcal{M}$ である。ユーザはスマートデバイス $\mathcal{U}_M$ を所有しているものとする。ここではユーザは各サービスシステムを利用するとき，PC などの端末 $\mathcal{U}_P$ を用いる[*5]ものとする。

[9] では，複数のサービスシステムが既に存在し，更にそのサービスシステムにユーザが登録されているという設定から始まり，新たに設置されたセンターに，各ユーザが自身のスマートデバイスを用いて登録するというシナリオであったが，ここでは，組織の全構成員 (ユーザ) がセンターに登録されており，各サービスシステムのユーザテーブルは空という初期状態から始まり，ユーザがスマートデバイスを用いて各サービスシステムを利用できるための登録作業を行うというシナリオで述べる。違いはユーザ情報の保存形態であり，実際 [9] においては，ユーザ情報を保有するのは各サービスシステムであり，センターは独自でユーザを特定できる情報を持たないという構成であるが，本論文では，図 4 に示す通り，センターがユーザ情報を統合的に管理し，各サービスシステムはユーザが利用するための最低限の情報しか保有しない。

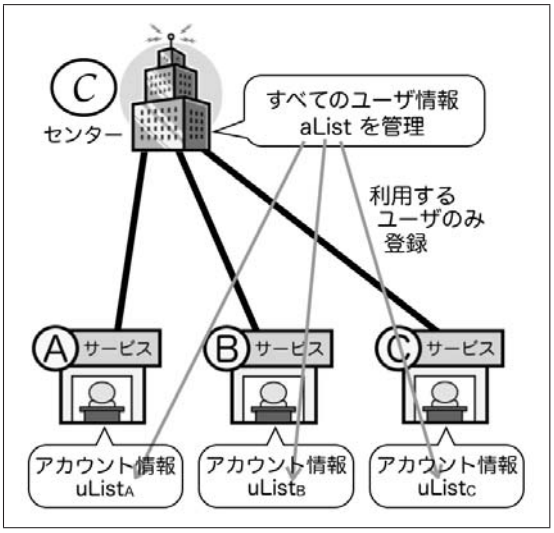

図 4　提案システムにおけるユーザ情報

3.1　初期状態

センター $\mathcal{C}$ には，組織構成員の情報が登録されているものとする。各ユーザに割り当てられる情報は，

(1) 利用者番号：uNo
(2) アプリケーション ID：aID
(3) リクエストパスワード：reqPw
(4) その他のユーザ属性：Att

である。リクエストパスワードは，システム導入時だとユーザに最初に割り当てられる初期パスワードのこととなるが，実際システムの利用を始めると，これは変更さ

[*5] 実際には，$\mathcal{S}$ もまたスマートデバイスで利用 (つまり $\mathcal{U}_P = \mathcal{U}_M$) しても構わない。

れ，当該ユーザしか知らない情報となる。

サービスシステム $\mathcal{S}$ には，まだユーザが登録されてなく，当該システムにサービスシステム ID(sysID) が割り当てられているだけとする。

ユーザが所有するスマートデバイス $\mathcal{U}_{\mathrm{M}}$ には，そこで動作する専用アプリ $\mathcal{A}$ が導入されているものの，$\mathcal{A}$ が $\mathcal{U}_{\mathrm{M}}$ 内に保有するデータベースは空である。

各ユーザは組織内で，システム管理者から利用者番号 (uNo)，リクエストパスワード (reqPw) およびメールアドレスなどの属性情報 (の一部) を受け取っているものとする。

3.2 登録手続き (その 1)

まず，ユーザ $\mathcal{U}$ はセンター $\mathcal{C}$ との間で登録手続きを行い，自身のスマートデバイス $\mathcal{U}_{\mathrm{M}}$(で動作する専用アプリ $\mathcal{A}$) を利用できる状態にする。

(R1) $\mathcal{U}$ は $\mathcal{A}$ を起動し，既に受け取っている利用者番号 (uNo) およびリクエストパスワード (reqPw) を入力し，それらを $\mathcal{C}$ に送る。

(R2) $\mathcal{C}$ は (uNo, reqPw) の正しさを検証した後，当該ユーザに対して，一意的なアプリケーション ID(aID) を割り当て，さらに aID およびその他のユーザ情報 info($\subset$ Att) に対して署名値 σ を生成，aID と併せたものをリクエスト依頼パス (Pass) とし，Pass を $\mathcal{A}$ に返す[*6]。

(R3) $\mathcal{A}$ は Pass を自身のデータベースに登録する。

以上の手続きを図 5 に示す。ここで Pass は，$\mathcal{U}_{\mathrm{M}}$ 固有の情報を鍵として暗号化された状態で $\mathcal{A}$ 内に保存されるものとする。このようにすることで，たとえ，データベースの情報が別のスマートデバイス $\mathcal{U}_{\mathrm{M}}{}'$ に不正に移されたとしても，$\mathcal{U}_{\mathrm{M}}{}'$ で動作する同一の専用アプリは Pass を正しく復号できない。つまり，正しい Pass を送ることができるのは，$\mathcal{U}$ がこの登録手続きで使用した $\mathcal{U}_{\mathrm{M}}$ で動作している $\mathcal{A}$ のみとなり，このことを利用して，$\mathcal{C}$ が $\mathcal{U}$ に対して所有物認証ができるようになる [2]。

3.3 登録手続き (その 2)

システム管理者 $\mathcal{M}$ が，ユーザ $\mathcal{U}$ がサービスシステム $\mathcal{S}$ を利用できるようにするための手続きである。システム管理者は，登録するユーザに対する利用者番号や，その権限などの情報 uInfo を知っているものとする[*7][*8]。

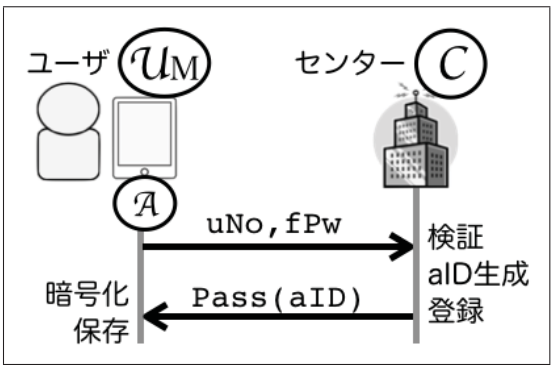

図 5　登録手続き (その 1)

(R4) $\mathcal{M}$ は，登録する ユーザ の uNo, uInfo および当該サービスシステムに割り当てられているサービスシステム ID(sysID) を $\mathcal{C}$ に送る[*9]。

(R5) $\mathcal{C}$ は，当該ユーザに対する uNo, uInfo および当該サービスシステム ID(sysID) に対する署名値 σ を，uNo, uInfo, sysID と併せて登録チケット (ticket) として，ticket を $\mathcal{U}$ に送る[*10][*11]。

(R6) $\mathcal{U}$ は $\mathcal{A}$ を使って ticket を $\mathcal{S}$ に送る。

(R7) $\mathcal{S}$ は ticket を $\mathcal{C}$ に送る。

(R8) $\mathcal{C}$ は ticket の正しさを検証した後，当該 (uNo, sysID) に対して一意的な管理 ID(mID) を生成し，mID を $\mathcal{S}$ に送る。

(R9) $\mathcal{S}$ は当該ユーザに対して一意的なユーザ ID(uID)，および $\mathcal{A}$ との共有鍵 suKey を生成し，uID, mID, uInfo, suKey を自身のデータベースに登録する。

(R10) $\mathcal{S}$ は，自身に関するシステム情報 sysInfo と共に，uID, suKey を $\mathcal{A}$ に返す。

(R11) $\mathcal{A}$ は ticket に含まれている sysID と (sysInfo, suKey) を自身のデータベースに登録する。

[*6] つまり，Pass = (aID, σ)，σ = Sign(aID, info) である。Pass は認証手続き (A2) において，その正しさを検証されるが，生成・検証はともに同一のエンティティ $\mathcal{C}$ が行うため，署名生成関数 Sign は公開鍵に基づくものである必要はなく，ハッシュ関数などを用いた MAC(Message Authentication Code) でも構わない。

[*7] uInfo は Att の部分集合，または，それから導かれる属性値とする。

[*8] インターネット上の一般的な一般的なサービスシステムであれば，ユーザ自身がシステム管理者の役割を担うことが多くなると思われるが，大学組織などの場合は，構成員に与えられる権限が多種に及ぶので，この手続きは，情報基盤を管理している部局など，特定の事務組織がシステム管理者を務めることになるであろう。

[*9] この作業を行う $\mathcal{M}$ は $\mathcal{C}$ に正しく認証されているものとする。

[*10] 具体的には ticket は (uNo, uInfo, sysID, σ) であり，その第 4 要素 σ は σ = Sign(uNo, uInfo, sysID) である。登録手続き (その 1) における (R2) に登場する Pass と同様，ticket を生成する際に用いられるアルゴリズム Sign もまた，公開鍵に基づくものである必要はない。

[*11] この ticket は，$\mathcal{C}$ から $\mathcal{A}$ に対するプッシュ通知により $\mathcal{A}$ が取得することもできるが，$\mathcal{U}$ が $\mathcal{A}$ を操作して $\mathcal{C}$ から取得することも可能である。ここでは，ticket が正しく $\mathcal{A}$ に送られるのであれば，その方法は特に指定しない。

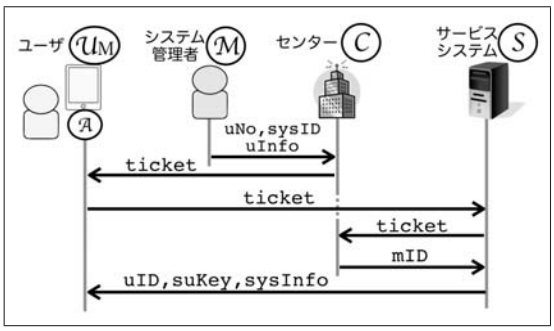

図 6　登録手続き (その 2)

以上の手続きを図 6 に示す。この手続きにより，各ユーザには 3 つの ID 情報 aID, mID, uID が割り当てられ，aID は $\mathcal{U}(\mathcal{A}) \Leftrightarrow \mathcal{C}$ 間，mID は $\mathcal{C} \Leftrightarrow \mathcal{S}$ 間，uID は $\mathcal{S} \Leftrightarrow \mathcal{U}$ 間で共有される。$\mathcal{U}$ 自身には aID および uID が通知されることになるが，それらを記憶する必要はない。実際，aID は $\mathcal{A}$ に保存されており，uID は $\mathcal{U} \Leftrightarrow \mathcal{C}$ 間の 2 要素認証が成功したことを通知された $\mathcal{S}$ から (suKey で暗号化された状態で)$\mathcal{A}$ に送られてくるためである。詳細については第 3.4 節で述べる。

1 人のユーザが複数のサービスシステムに登録される場合は，登録手続き (その 2) をその分だけ実行することになる。それにより，センター $\mathcal{C}$ 内の当該ユーザ $\mathcal{U}$ には，$\mathcal{U}$ が登録されたサービスシステムの個数分だけ mID が紐付けられ，専用アプリ $\mathcal{A}$ には同じ個数分だけ (sysID, sysInfo, suKey) が登録されることになる。

3.4　認証手続き

本提案システムにおける認証手続きは，[9] と同様のプロセスを経るものであり，図 3 がそのまま通用する。ただし，図 3 における (1) のリクエストは「ワンタイムパスワード発行依頼」となり，(2) においてセンターはワンタイムパスワード (OTP) を生成し，当該ユーザの識別子と併せてサービスシステムに送る。(4) において，センターは OTP をユーザに通知し，ユーザは (5) において，通知された OTP を用いてサービスシステムにログインする。以下に，サービスシステム $\mathcal{S}$ に登録されたユーザ $\mathcal{U}$ が $\mathcal{S}$ にログインするための手続きを述べる。

ユーザ $\mathcal{U}$ はスマートデバイス $\mathcal{U}_{\mathrm{M}}$ を所持しており，$\mathcal{U}_{\mathrm{M}}$ で動作する専用アプリ $\mathcal{A}$ には，アプリケーション ID　aID を含むリクエスト依頼パス Pass が保存されている。

(A1) $\mathcal{U}$ は $\mathcal{A}$ にリクエストパスワード reqPw を入力し，利用したいサービスシステムを選択し，そのサービスシステム ID(sysID) を reqPw, Pass と共に $\mathcal{C}$ に送る。

(A2) $\mathcal{C}$ は，Pass の正しさを検証し，それに含まれる aID から当該ユーザを特定する[*12]。また reqPw の正しさを検証し，正しい場合は，ワンタイムパスワード otp を生成し，当該ユーザに割り当てられている mID と共に $\mathcal{S}$ に送る[*13]。

(A3) $\mathcal{S}$ は mID に紐付けられている uID および suKey を特定し，その uID に対して otp およびその有効期限 period を設定する。その後，uID を suKey を鍵として暗号化したものを $E_{\mathsf{suKey}}(\mathsf{uID})$ とし，それを period と合わせたもの $\mathsf{res} = (\mathsf{period}, E_{\mathsf{suKey}}(\mathsf{uID}))$ を $\mathcal{C}$ に返す。

(A4) $\mathcal{C}$ は (otp, res) を $\mathcal{A}$ に返す。

(A5) $\mathcal{A}$ は res に含まれる $E_{\mathsf{suKey}}(\mathsf{uID}))$ を復号し得られた uID を (otp, period) と共に表示し，$\mathcal{U}$ に知らせる。

(A6) $\mathcal{U}$ は，period で定められた期限内に (uID, otp) を $\mathcal{S}$ に送りログインする[*14]。$\mathcal{S}$ は uInfo に基づいた権限で $\mathcal{U}$ にサービスを提供する。

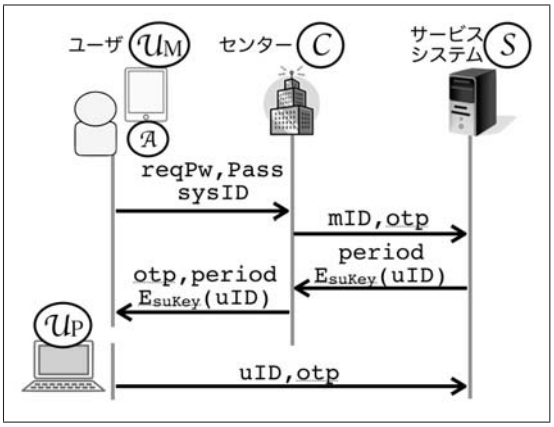

図 7　認証手続き

以上の手続きを図 7 に示す。ユーザが記憶する必要があるのは reqPw のみであることに注意されたい。ユーザ $(\mathcal{U}_{\mathrm{M}}(\mathcal{A}), \mathcal{U}_{\mathrm{P}})$ から発信する情報で reqPw 以外のものは全て $\mathcal{A}$ に保存されているか，そうでなければ，$\mathcal{S}$ から $\mathcal{C}$ 経由で送られてきたもの (または，それを復号したもの) であり，$\mathcal{U}_{\mathrm{M}}(\mathcal{A})$ に表示される。

また，uID は $\mathcal{A} \Leftrightarrow \mathcal{S}$ 間で共有された鍵で暗号化された状態で $\mathcal{C}$ を通過するため，$\mathcal{C}$ は uID を知り得ない。そ

[*12] 例えば，大学組織において $\mathcal{U}$ が「卒業」や「退職」するなどしてアクセスする権限を失っていた場合，$\mathcal{C}$ 内のユーザ情報を更新することにより，この段階でリクエストを拒否することができる。

[*13] $\mathcal{S}$ は，$\mathcal{A}$ から送られてくる sysID により特定できる。

[*14] 安全性を強化する場合は，(A6) において $\mathcal{U}$ が (uID, otp) の他に reqPw も送るようにし，それを受け取った $\mathcal{S}$ が (mID, reqPw) の正しさを $\mathcal{C}$ に照会することにより，「ユーザ ID，(固定) パスワード，ワンタイムパスワード」で認証を行えばよい。本システムにおいては，ワンタイムパスワードを入手するために ($\mathcal{U} \Leftrightarrow \mathcal{C}$ 間で) 記憶認証を行っているので，$\mathcal{U} \Leftrightarrow \mathcal{S}$ 間の記憶認証は省略している。

れは，たとえ $\mathcal{C}$ が悪意のある者に操作され，不正にワンタイムパスワードを発行されたとしても，当該 mID に対応する uID が解らない[*15]ため $\mathcal{S}$ への不正ログインを阻止できることを意味する。

さらに，$\mathcal{U}$ が (A1) を実行しない限り $\mathcal{S}$ にワンタイムパスワードが設定されないため，如何なる文字列を otp として送ったとしても，ログインは拒否される。これは提案システムが，[9, 10] で述べられている「認証シャッター」機能を備えていることを意味する。

4 考察

本章では，提案システムの安全性と利便性について考察する。

4.1 安全性

まず，第三者 $\widetilde{\mathcal{U}}$ が特定のユーザ $\mathcal{U}$ になりすますことを考える。$\widetilde{\mathcal{U}}$ が $\mathcal{U}$ としてサービスシステム $\mathcal{S}$ にログインするためには，$\mathcal{S}$ に保存されている $\mathcal{U}$ の情報 uID だけでなく，$\mathcal{U}$ がセンター $\mathcal{C}$ にワンタイムパスワード発行依頼 (手続き (A1) を実行) したときに決定した otp をそのとき $\mathcal{S}$ により定められた period の期限内に入手しなければならない。$\mathcal{U}$ が発行依頼をしない限り，如何なる otp を $\mathcal{S}$ に送ったとしても，$\mathcal{S}$ の認証シャッター機能により拒否される。また，正しい uID, otp を入手するためには，$\mathcal{U}$ が所持するスマートデバイス $\mathcal{U}_{\mathrm{M}}$，および，$\mathcal{U}$ の記憶情報 reqPw が必要となるので，これは一般的に困難であると思われる。

次に $\mathcal{S}$ に内部不正者がいた場合を考える。登録されているユーザになりすましてログインするためには，ワンタイムパスワードの設定が正しく行われなければならない。そのためには前述と同様，$\mathcal{U}_{\mathrm{M}}$ および reqPw が必要となる。

$\mathcal{C}$ に内部不正者がいた場合を考える。$\mathcal{C}$ は接続されている任意のサービスシステム $\mathcal{S}$，$\mathcal{S}$ に登録されている任意のユーザ $\mathcal{U}$ に対してワンタイムパスワード otp を設定できる。しかし，どの (sysID, mID) に対しても，その $\mathcal{S}$ にログインするための (mID に紐付けられた)uID を知らないため，不正ログインはできない。

4.2 利便性

ワンタイムパスワード認証システムの場合，どのようにしてユーザに otp を取得させるかによって状況が変わるが，2 要素認証になっている場合，ユーザの負担は「記憶情報，所有物」に依る。乱数表方式やハードトークン方式の場合は，明らかに，所有物に起因する負担がかかる。これを解決しているのがソフトトークン方式である。しかし，どの方式であったとしても，最低限「ユーザ ID と (固定) パスワード」の 2 つは記憶する必要があり，しかも，利用するサービスシステムが増えるにつれて記憶しなければならない情報は増加する。一方，本提案システムでは，覚えなければならない情報を「リクエストパスワード」の 1 つに軽減しており，これは利用するサービスシステムの数に依存しない。

4.3 Man in the browser (MITB) 対策可能性

本提案システムはユーザ $\mathcal{U}$，サービスシステム $\mathcal{S}$ およびセンター $\mathcal{C}$ の 3 者による 2 経路認証の形になっている。そのため，本提案システムを拡張することにより，ログイン認証だけではなく，ログイン後に行う処理に対する認証も可能となる。例えば，$\mathcal{U}$ が $\mathcal{S}$ に送ったリクエスト内容を，実行確認として，$\mathcal{C}$ 経由で $\mathcal{U}_{\mathrm{M}}(\mathcal{A})$ に通知することにより，仮に $\mathcal{U}_{\mathrm{P}}$ 内に潜むウイルスによりリクエスト内容を改ざんされていたとしても，$\mathcal{U}$ は $\mathcal{U}_{\mathrm{M}}$ の通知により検知でき，改ざんされたリクエストの実行を事前に阻止することができる。2 経路認証は，既に一部のネットバンキング等で利用されている仕組みであるが，将来的には大学組織にも必要とされる可能性がある。

5 まとめ

本論文では複数のサービスシステムに適用できるワンタイムパスワード認証システムでありながら，ユーザの負担を極力減らすことを目的としたシステムの構築方法を提案した。ネットワーク上のサービスシステムは今後統廃合を繰り返しながら，益々多種多様なサービスを提供するようになると思われる。サービスの性質によっては，(固定) パスワードによる認証では不十分なものが多く，記憶情報だけではなく，所有物 (SYH) や生体情報 (SYA) を利用した認証も必要になるであろう。現在は，そのような認証に馴染みがなかったり，そのような認証を必要と思うユーザの割合は少ないかもしれないが，実際，(固定) パスワードの安全性が問題視され，被害も頻繁に発生していることから，より強い安全性 (認証レベル) を満たし，かつ，ユーザの負担を考慮した認証システムの構築が望まれる。また，本論文はシステム内のプロトコル構築を主としており，具体的な実装方法等については触れていなかったので，提案システムを実装実験することにより，さらなる安全性および利便性の向上を目指すことを，今後の課題としておく。

謝辞

本研究の一部は JSPS 科研費 15K00181 の助成を受けたものです。また，本論文を執筆するにあたり助言をいただいた，株式会社セフティーアングルの糸井正幸氏に，この場を借りて心より感謝申し上げます。

[*15] mID と uID の対応を知っているのは $\mathcal{S}$ のみである。

参考文献

[1] IPA：オンライン本人認証方式の実態調査報告書, 2014.
https://www.ipa.go.jp/security/fy26/reports/ninsho/
(参照：2016 年 8 月 29 日)

[2] 垣野内，木下，多田，糸井，山岸：「プライバシーを考慮したワンタイムパスワード認証システムの実装」，2012 年暗号と情報セキュリティシンポジウム (SCIS2012), 1E2-5, 2012.

[3] 関西大学 IT センター：「利用者 ID とパスワード」,
http://www.itc.kansai-u.ac.jp/start/idpw.html
(参照：2016 年 8 月 29 日)

[4] 慶應義塾 ITC：「keio.jp におけるワンタイムパスワードについて」,
http://www.itc.keio.ac.jp/ja/keiojp_otp.html
(参照：2016 年 8 月 29 日)

[5] 神戸学院大学情報支援センター：「学内システム一覧，学内情報サービス (在学生・教職員対象)」,
http://www.kobegakuin.ac.jp/facility/ipc/system-lst.html
(参照：2016 年 8 月 29 日)

[6] L. Lamport: *Password authentication with insecure communication*, Communications of the ACM, vol.24, no.11, pp.770-772, 1981.

[7] 南山大学：「学外からの使いかた (Can@home)」,
http://office.nanzan-u.ac.jp/CAN/can_usage.html
(参照：2016 年 8 月 29 日)

[8] 佐賀大学総合情報基盤センター：「シングルサインオン、セキュリティ強化のための多要素認証導入」,
http://www.cc.saga-u.ac.jp/plan/webnews.php?num=365
(参照：2016 年 8 月 29 日)

[9] 多田：「パスワード認証の強化策」，学術情報処理研究, no.19, pp.40-49, 2015.

[10] 高田：「Authentication shutter: 個人認証に対する攻撃を遮断可能する対策の提案」，コンピュータセキュリティシンポジウム (CSS)2014, 3C1-3, 2014.