九州大学学術情報リポジトリ
Kyushu University Institutional Repository

User-Computer間の認証に関する考察

渡部, 貴大
九州大学大学院システム情報科学府

野原, 康伸
九州大学大学院システム情報科学府

馬場, 謙介
九州大学大学院システム情報科学府

井上, 創造
九州大学大学院システム情報科学府

他

https://hdl.handle.net/2324/6263

出版情報：SLRC 論文データベース, 2006-01
バージョン：
権利関係：

All rights are reserved and copyright of this manuscript belongs to the authors. This manuscript has been published without reviewing and editing as received from the authors: posting the manuscript to SCIS 2006 does not prevent future submissions to any journals or conferences with proceedings.

SCIS 2006 The 2006 Symposium on Cryptography and Information Security
Hiroshima, Japan, Jan. 17-20, 2006
The Institute of Electronics, Information and Communication Engineers

User-Computer 間の認証に関する考察

渡部　貴大* 野原　康伸* 馬場　謙介* 井上　創造*
Takahiro Watanabe Yasunobu Nohara Kensuke Baba Sozo Inoue
安浦 寛人*
Hiroto Yasuura

あらまし Electronic authentication with a portable device such as a smart card has been receiving increasing attention. In an authentication, the portable device is regarded as the human user himself. However, in an open environment like authentication systems, it is necessary to have a way of secure communication between the portable device and the human user. This paper considers an authentication of a server computer of a service provider by a human with a portable device as a part of the authentication and an attack by a client computer which relays the communication between the portable device and the server computer. As a defense against the attack, we introduce a system with a portable device which has an interface to show information to a human.

キーワード 認証トークン, 相互認証, リンク不能性

1 はじめに

近年，電子商取引など重要な情報を扱う電子サービスの増加に伴い，第三者によるユーザへのなりすましやサーバコンピュータへのなりすまし（フィッシング詐欺など）が問題になっている．特に遠隔地のコンピュータを利用する場合，ユーザはサービスを視覚的に確認することが出来ないため非常に困難であるためなりすましの被害にあう可能性が高い．これらのなりすましを防止するためには，ユーザとコンピュータの間で相互に認証を行う必要がある．

コンピュータによるユーザの認証方法としては，秘密情報を共有し，人間の処理能力で利用可能な演算を用いて安全に認証を行う手法が提案されている [1][2]．一方で，ユーザがコンピュータを直接認証する手法はほとんど議論されていない．

ユーザがコンピュータを認証する手法の一つとして，IC カードなどの認証トークンを用いた相互認証を利用する技術が提案されている [3][4]．高い計算能力を備えた認証トークンを利用することによって，トークンとサーバコンピュータの間でゼロ知識証明によって証明された認証や，ワンタイムパスワードなどの安全な相互認証を行うことができる [5][6]．

しかし，トークンを用いてユーザとコンピュータの間で相互認証を行う場合，トークンとコンピュータの間の安全性以外に以下の 2 点を考慮する必要がある．

まず，トークンを紛失・盗難した際に持ち主以外の人間がトークンを利用して認証を成功させることを防がなければならない．そのため，バイオメトリクスなどの技術を利用し，コンピュータがトークンとユーザを関連付けることによりトークンの持ち主でなければ認証が行えないようにする必要がある．

また，ユーザとトークン間の通信についても考える必要がある．ユーザが IC カードのようなユーザに直接情報を伝えるインタフェースを持たないトークンを利用する場合，トークンによる認証結果はクライアントコンピュータ（以下，クライアント）に表示されることになる．その場合，クライアントは認証結果を改ざんしユーザを騙す攻撃が可能になる．我々が想定する攻撃の例を以下に示す（図 1）．

1. トークンはサーバコンピュータを偽者と認証する
2. トークンはクライアントに “認証失敗” と伝える
3. クライアントはユーザを騙すために，“認証成功” とユーザに伝える

上記の攻撃は，トークンがサーバコンピュータを正しく認証していてもクライアントはユーザに対する攻撃を成功することが出来ることを示している．これは，トークンがコンピュータを認証することはユーザがコンピュー

* 九州大学大学院システム情報科学府 〒 816-8580 春日市春日公園 6-1 E 棟 I403 Kyushu University 6-1 Kasugakoen, Kasuga-City, Fukuoka 816-8580, JAPAN takahiro@c.csce.kyushu-u.ac.jp

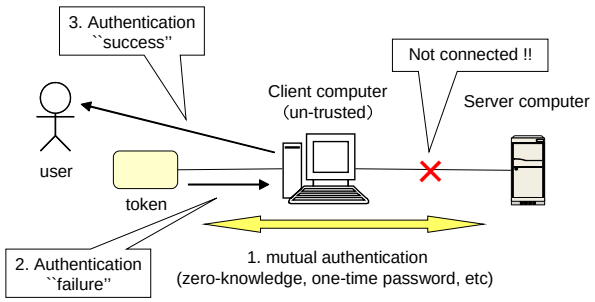

図 1: クライアントコンピュータによる攻撃

タを認証することと同じではないということを意味する．クライアントが信用できない場合，トークンとコンピュータの間の認証を安全に行うだけでなく，クライアントからの攻撃を想定したユーザとトークンの間の認証を安全に行う技術についても考える必要がある．

本論文の構成は以下のとおりである．2 章でクライアントによる認証結果の改ざんという攻撃を，認証プロトコルをユーザとトークンを分離してモデル化することにより明確にする．3 章では，クライアントによる攻撃をユーザインタフェースを持つトークンが防止できることを示す．4 章では，ユーザ名を用いた認証を提案しユーザ間のリンク不能性を満たすことによってクライアントの攻撃を防止できることを示す．5 章でまとめる．

2 モデル化

本章では，筆者が想定する認証システムの構成をモデル化することによって説明する．

認証モデルは以下の 4 つの主体から構成される．

ユーザ $\{u_i\}(1 \leq i \leq N)$ ：サーバを利用する人間．トークン (t_i) を所持している．

サーバ $\{s\} \in \{s_r, s_f\}$ ： ユーザにサービスを提供するサーバコンピュータ．s_r は正規サーバ．s_f は偽サーバ．

クライアント c ：サーバ，トークン間の通信を中継するコンピュータ．ユーザに情報を表示するためのディスプレイがある．

トークン $\{t_i\}(1 \leq i \leq N)$ ：サーバと相互認証を行うデバイス．ユーザ (u_i) によって信頼されている．

また各主体間を流れる情報を $r_{x \to y} = \{accept, reject\}$ $(x, y \in \{u_i, t_i, c, s\})$ と定義する．トークンによるサーバの認証が成功した場合は $accept$，失敗した場合は $reject$ である．

次に，各主体の関係を以下に示す（図 2）．

- サーバはクライアントにデータを送信できる
- トークンはクライアントにデータを送信できる
- ユーザはクライアントにデータを送信できる
- クライアントはサーバ，トークンにデータを送信し，ユーザにデータを表示できる

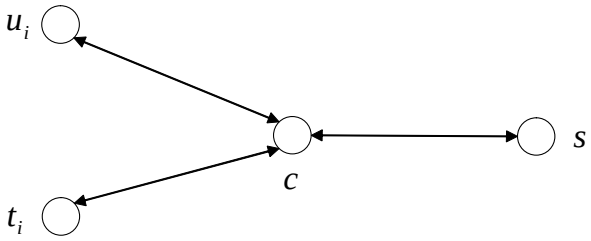

図 2: 各主体の関係

上記の各主体間の関係を基に，ユーザ $(u_i \in \{u_i\})$ とサーバ $(s \in \{s_r, s_f\})$ 間の相互認証は以下の手順で行われる (図 3)．

1. ユーザはクライアントにサーバ名を送信する
2. クライアントはユーザから要求されたサーバに接続を要求する
3. トークンとサーバの間で相互認証を行う
4. トークンはサーバに対する認証結果 $\{accept, reject\}$ をクライアントに送信する $(r_{t_i \to c} \in \{accept, reject\})$.
5. クライアントはユーザに認証結果を表示する $(r_{c \to u_i} \in \{accept, reject\})$.

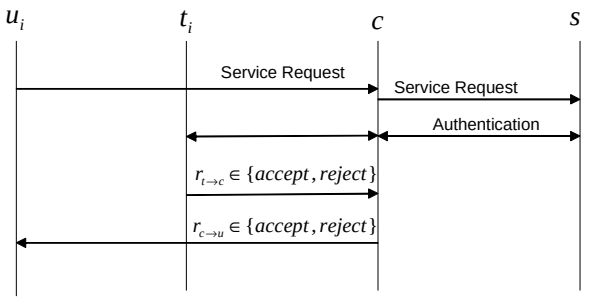

図 3: ユーザとサーバの間の認証プロトコル

この手順では，トークンとサーバの間の相互認証には，以下の条件を満たす認証方式が利用されていると仮定されている．

- s_r に自分が t_i であることを証明できるトークンは t_i のみである
- $t_i(1 \leq i \leq N)$ に自分が s_r であることを証明できるのは s_r のみである

サーバは，トークンを利用している人間が正しいユーザであるということをバイオメトリクスなどの技術を使い確認できれば，ユーザを認証したと言える．

本論文では，トークンによるサーバの認証は，偽サーバによるなりすましを防止することを目的とする．そのため，偽サーバを正規サーバとする（$r_{t_i \to c} = reject$ を $r_{c \to u_i} = accept$ とする）クライアントのみ攻撃者であると定義し，正規サーバをトークンが $accept$ と認証した場合に，クライアントが表示する情報が $reject$ となることは攻撃とみなさないとする．

表 1: クライアントによる攻撃

s	s_r		s_f	
$r_{t_i \to c}$	$accept$	$accept$	$reject$	$reject$
$r_{c \to u_i}$	$accept$	$reject$	$reject$	$accept$
c				攻撃者

この認証モデルにおいて，トークンは正しい認証結果を送信する．そのため，クライアントから表示される認証結果がトークンからの認証結果と同じであることをユーザが確認できる場合，攻撃を防止することが出来る．

また，クライアントによる攻撃は認証結果の改竄であるため，クライアントが認証結果を改竄できなければ攻撃は成功しない．

以上より，以下の 2 つの条件のうちどちらか一つを満たすことによりクライアントからの攻撃を防止することが出来る．

条件 1 クライアントは，“$r_{t_i \to c} = reject$” のとき “$r_{c \to u_i} = accept$” とできない

条件 2 ユーザは，“$r_{t_i \to c}$” を確認できる

3 認証可能トークン

本章では，ユーザインタフェースつきのトークンを用いた認証について述べる．

ユーザインタフェースつきトークンを表すために，前節で述べた各主体間の関係に以下の一つを追加する．

- トークンはユーザにデータを表示出来る

この関係を付加した各主体間の関係を図 4 に示す．

各主体間が図 4 のように接続されているとき，ユーザ ($u_i \in \{u_i\}$) とサーバ ($s \in \{s_r, s_f\}$) の相互認証は，以下の手順で行うことが出来る (図 5)．

1. ユーザはクライアントにサーバ名を送信する
2. クライアントはユーザから要求されたサーバに接続を要求する

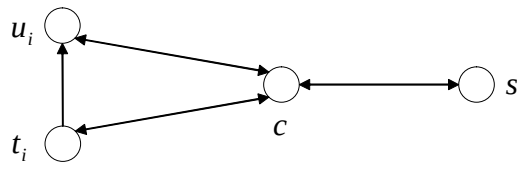

図 4: トークンにユーザインタフェースがある場合の各主体の関係

3. トークンとサーバの間で相互認証を行う
4. トークンは認証結果をユーザに提示する

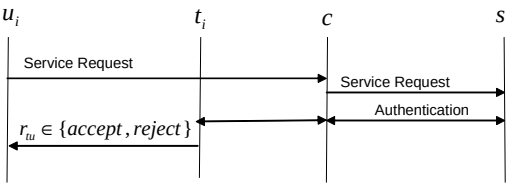

図 5: ユーザインタフェースつきトークンを用いた認証プロトコル

本論文では，トークンはユーザにとって信頼できるデバイスであると定義しているため，トークンが提示する認証結果は正しいと言える ($r_{t_i \to u_i} = r_{t_i \to c}$)．よってユーザインタフェースを持つトークンを用いたユーザとサーバの間認証は正しいと言える．

4 ユーザとトークン間の認証方法の検討

前章ではユーザインタフェースを持つトークンを認証に利用することにより，クライアントコンピュータからの攻撃を防ぐことが出来ると述べた．しかし現実には，IC カードのようなユーザインタフェースを持たないトークンが多数存在する．本章では，ユーザインタフェースを持たないトークンがクライアントからの攻撃を防ぐ方法について検討する．

4.1 ユーザ名を用いた認証

2 章でモデル化したシステムに以下の情報を追加する．

ユーザ名 $\{a_i\}(1 \leq i \leq N)$ ：ユーザが記憶できる長さの文字列．トークン (t_i) に記憶されている．

ユーザ ($u_i \in \{u_i\}$) とサーバ ($s \in \{s_r, s_f\}$) の相互認証は以下の手順で行われる (図 6).

1. ユーザはクライアントにサーバ名を送信する
2. クライアントはユーザから要求されたサーバに接続を要求する

3. トークンとサーバの間で相互認証を行う

4. トークンはクライアントに対して認証成功ならば a_i，認証失敗ならば $reject$ を送る.($r_{t_i \to c} \in \{a_i, reject\}$)

5. クライアントはユーザに認証結果を表示する.($r_{c \to u_i} \in \{a_i, reject\}$)

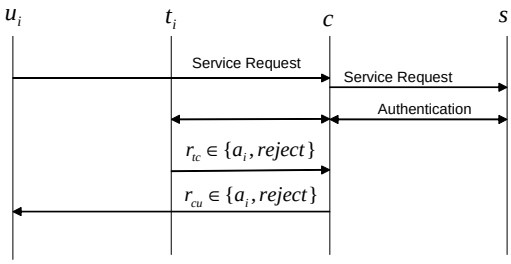

図 6: ユーザ名を用いた認証プロトコル

この認証システムにおいて，ユーザ (u_i) はクライアントにユーザ名 (a_i) が表示されればトークンによる認証は成功したと判断する．つまり，トークンによる認証結果が $reject$ であっても，クライアントに表示される認証結果がユーザ名 (a_i) であればユーザは認証が成功したと判断し，クライアントによる攻撃は成功する．

表 2: クライアントによる攻撃

s	s_r		s_f	
$r_{t_i \to c}$	a_i	a_i	$reject$	$reject$
$r_{c \to u_i}$	a_i	$reject$	$reject$	a_i
c				攻撃者

4.2 攻撃防止の条件

クライアントがユーザ (u_i) に攻撃を行うためには，以下の 2 つの条件を満たす必要がある．

1. クライアントは a_i を知ることが出来る

2. クライアントは認証を行っているユーザが過去に通信を行ったユーザと同一人物であることが分かる

特に 2 の条件をクライアントが満たすことを防ぐことを，クライアントに対してユーザ間のリンク不能性 (Unlinkability) を満たすということが出来る．リンク不能性は，ユーザが複数の資源あるいはサービスを使用するとき，他人がそれらを一つにリンクできないようにして使用できることを保証する性質であると定義されている [7].

4.3 攻撃

前節で述べた 2 つの条件を満たすことが出来るか検証するため，ユーザとサーバの間認証における正規のクライアントの動作手順を以下に整理する．

手順 1 ユーザから接続するサーバ名を受け取る

手順 2 手順 1 のサーバに接続する

手順 3 トークンと接続する

手順 4 トークンとサーバの間の相互認証を中継する

手順 5 トークンから認証結果を受け取る

手順 6 ユーザに認証結果を表示する

まず，1 つ目の条件を満たす方法について検討する．クライアントは手順 1 において正しいサーバと接続しトークンによるサーバの認証を成功させることで，手順 5 においてトークンからユーザ名を取得できる．よって，正しいサーバに接続することが可能なクライアントであれば，1 の条件を満たすことが出来てしまう．

次に，2 つ目のユーザ間のリンク不能性を破る方法について検討する．ユーザ間のリンク不能性を破るためにはユーザを識別する情報が必要である．クライアントの動作手順の中で，ユーザを識別するための情報が取得できる可能性があるのは，トークンから情報を受け取ることができる手順 3，手順 4，手順 5 の 3 つである．

手順 3 では，トークンとの接続時にトークン固有の ID などトークンを識別する情報が送信される場合がある．また手順 4 では相互認証中にトークンが送信する情報が毎回同じであればトークンを識別することが可能である．一つトークンのユーザは一人なので，トークンを識別することが出来ればユーザを識別することが出来る．

しかし，手順 3 についてはトークンが ID の送信を行わないプロトコルにすることでトークンの識別を防止することが出来る．また，手順 4 については Randomized Hash Lock 方式 [8] や Extended Hash-chain 方式 [9] など認証ごとに異なる情報を送信する認証をトークンとサーバの間で行うことにより，トークンの識別を防ぐことが出来る．

また，手順 5 では正規のサーバに接続する認証では a_i を知ることが可能であるため，ユーザ間のリンク不能性を破ることが出来る．しかし，クライアントが攻撃を行うためには偽サーバに接続する認証においてユーザ間のリンク不能性を破らなければならない．偽サーバに接続した場合，手順 5 において受け取ることが出来るのは $reject$ のみであるため，リンク不能性を破ることは出来ない．

ユーザ間のリンク不能性が実現されている場合，クライアントは全てのユーザのユーザ名 $\{a_i\}(1 \leq i \leq N)$ を

記憶しているとしても，攻撃が成功する確率は $1/N$ である．N が十分に大きい場合このプロトコルはクライアントコンピュータからの攻撃を防ぐことが出来るといえる．

また，クライアントがユーザ名が分からない場合に攻撃が成功する確率は，a_i の文字列の長さに依存する．

4.4 リンク不能性に対する攻撃

前節では，ユーザ間のリンク不能性をクライアントに破られなければクライアントによる攻撃を防ぐことが出来ると述べた．本節ではクライアントがリンク不能性を破る可能性がある攻撃方法を二つ紹介する．

- ユーザ情報による識別

  ユーザ情報の識別による攻撃は，クライアントにカメラなどを設置し，認証を行うユーザの画像情報を記憶する．記憶する画像情報が過去の認証において記憶したユーザの情報と一致した場合，ユーザ間のリンク不能性を破ることが出来る．

- 時間制限による識別

  時間制限による識別を行う攻撃では，同じクライアントで行われる認証の中でほぼ同じ時間に行われる認証を同一ユーザによる認証であると推測する方法である [10].

  クライアントの動作の一例を以下に示す．

  1. ユーザの要求するサーバと接続し，トークンから a_i を取得
  2. サーバとの接続を故意に切断する
  3. 次に同じサーバとの接続を要求するユーザを u_i と識別する
  4. ユーザの要求とは異なるサーバに接続する
  5. トークンから *reject* を受信する
  6. ユーザに a_i を表示する

  この場合，ユーザがクライアントの意図どおり認証を要求すれば，リンク不能性は破られる．

5 おわりに

本論文では，信用できるトークンを利用したユーザとサーバの間の認証をユーザとトークンを分離してモデル化し，クライアントによる攻撃の可能性と攻撃を防止する条件について述べた．また，表示機能などのユーザインタフェースを持つトークンを使用すれば，攻撃を防止する条件を満たすことが可能であることを示した．そのため，IC カードなど現時点でユーザインタフェースを持たないトークンが表示機能をつけること（LED など 1 ビットの情報だけでもよい）は攻撃を防止することにつながるため有用であるといえる．

また，本論文ではユーザインタフェースを持たないトークンを用いた認証プロトコルとして，ユーザ名を送信するプロトコルについて検討した．ユーザごとに異なるユーザ名を認証成功時に送信するこのプロトコルを用いる場合，認証を行うユーザ間のリンク不能性を満たすことによって攻撃を防止することができる．

しかし 4.4 節で述べた攻撃方法により，リンク不能性を破ることができる可能性がある．今後はこれらの攻撃がどこまでリンク不能性を破ることができるか検証する．また，これらの攻撃に対してもリンク不能性を守るプロトコルについて考えていく．

謝辞

本研究は，平成 14 － 18 年度科学研究費補助金学術創成研究・課題番号 14GS0218 によるものである．

参考文献

[1] Hopper, N.J. and Blum, M., “Secure Human Identification Protocols”, ASIACRYPT 2001, LNCS, vol.2248, pp.52–66, 2001.

[2] Matsumoto, T. and Imai, H., “Human Identification Through Insecure Channel”, EUROCRYPT 91, LNCS, vol.547, pp.409–421, 1991.

[3] Naomaru Itoi, Peter Honeyman., “Smartcard integration with Kerberos V5”, In USENIX Workshop on Smartcard Technology, Chicago, May 1999.

[4] Bastiaan Bakker. “Mutual authentication with smart cards” In USENIX Workshop on Smartcard Technology, Chicago, May 1999.

[5] Fiat, A. and Shamir, A., “How to prove yourself: Practical solutions to identification and signature problems”, CRYPTO ’86, LNCS, vol.263, pp.186–194, 1987.

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

[7] “ISO/IEC 15408 - INTERNATIONAL STANDARD Information technology - Security techniques - Evaluation criteria for IT security - Part2: Security functional requirements”, Dec.1999.

[8] Stephan A. Weis, Sanjay E. Sarma, Ronald L. Rivest, and Daniel W. Engels,“ Security and Privacy Aspects of Low-Cost Radio Frequency Identi-

fication Systems ”, International Conference on Security in Pervasive Computing 2003, LNCS 2802, pp.201-212, 2004.

[9] 大久保美也子, 鈴木幸太郎, 木下真吾,“ Hash-Chain Based Forward-Secure Privacy Protection Scheme for Low-Cost RFID ”, SCIS2004 論文集, 2004 年 1 月

[10] Yasunobu Nohara, Sozo Inoue, and Hiroto Yasuura, “Toward unlinkable ID Management for Multi-service Environments”, Proc. 3rd Int’l Conf. Pervasive Computing and Communications(PerCom) Workshops, pp.115-119, Mar. 2005.