

スマートカードを用いた認証システムの開発と インターネットへの応用

鈴木 利則 大橋 正良 渡辺 文夫

国際電信電話（株） 研究所

筆者らは、移動通信網やインターネットのような容易にアクセス可能なネットワークの普及と、それに伴う付加価値の高い通信サービスの登場を指摘するとともに、このような環境下で必要とされる認証方式を提案、試作してきた。本稿では、試作した認証方式について紹介するとともに、これをインターネット上のWWWアプリケーションに組み合わせて実装した例について報告する。

Development of Authentication Mechanism using Smart Cards and Its Application onto the Internet

Toshinori Suzuki Masayoshi Ohashi Fumio Watanabe

KDD R&D Laboratories

The authors have pointed out that the spread of network which could be easily accessed, such as the mobile communications network or Internet, would stimulate the requirements for more value-added services. In such environment, more secure authentication mechanism will be required. That's our aim to have developed the authentication mechanism which is independent on the terminal's security. This paper reports our developed authentication system which makes use of smart cards. Additionally we also report its implementation in combination with WWW application onto the Internet.

1 はじめに

筆者らは、移動通信網やインターネットのような不正なユーザが容易にアクセス可能なネットワークの普及と、それに伴う付加価値の高い通信サービスの登場を指摘するとともに、このような環境下で必要とされる認証方式として、端末に依存しない安全性を確保するとともに相互認証を採用した認証方式を提案、試作している[1,2]。

本稿では、既に試作している同認証システムについて簡単に紹介するとともに、これを応用した、WWWサーバと連携した認証機能付きファイル転送ソフトウェアを作成したので報告する。

2 認証メカニズム

2.1 背景

インターネットはその公開性を背景として、広く普及してきた。このような利用形態、すなわち誰もが情報発信者になりうるし、誰もがその受け手であるような形態、は引き続き広まってゆくであろう。

一方においてそのオープンなネットワーク上にありながら、閉じた利用形態も顕在化していくものと考えられる。例えば、会員制の情報提供サービスやオンラインショッピング、学内/企業内LANなどである。現状ではセキュリティの問題から、例えばオンラインショッピングで決済をクレジットカードで行う場合、そのリスクを考えて手数料を高めめに設定する、利用金額の上限を小さくするとか、企業内LANにしても重要な情報はネットに流さないなどの処置を施していることが多く、まだその利用形態に種々の制約が課せられているようである。

より安全な認証方法を開発することは、このような閉じた利用形態の制約を緩和し、付加価値の高い通信サービスを実現するために必須である。

2.2 従来の認証方式

従来の認証方式としてKerberosが有名である。これはパスワードの転送を暗号化することで、タッピングからの脅威を防いだものであるが、いったんパスワードがわかってしまえばあとは容易に偽装することができる。パスワードは認証の度に露呈するものであるし、推測可能であることや、端末に仕掛けられうる不正ソフトの脅威から免れないなどの問題が指摘されている[3]。

またデバイスを使ったインターネット上の認証方式として、スーパースマートカード(キーパッドによる入力手段と液晶による表示手段を有しているCPU付きICカード)を用いたセキュリティゲートウェイなどが販売されている。これはセキュリティゲートウェイが乱数(チャレンジ)を生成し、端末を通してユーザに表示する。ユーザはそれをスーパースマートカードに入力すると、ワンタイムパスワードをスーパースマートカードが提示する。それをユーザは端末に入力することで認証を行うものである。乱数のかわりに時刻などの情報を用いるものもある。

これらの方式はいずれもリモートの環境でユーザに認証する手段を提供するもので、クライアント側にはリーダ・ライタなどの特別なハードウェアはいらないが、ユーザが直接認証情報(チャレンジやワンタイムパスワード)をスーパースマートカードまたは端末に入力しなければならないので、ユーザに負担を強いることは否めないし、短い間隔で認証を要求するような形態には適さない。

2.3 スマートカードの利用

本認証システムではユーザ認証用のデバイスとしてスマートカードを用いる。閉じたグループでは、例えば会員/社員/学生がその証としてスマートカードを有している状況は十分に現実的である。

筆者らはスマートカードを積極的に利用することで、以下の狙いを考えている。

- 1) 認証時にユーザに強いる手続を軽減。
- 2) 多様な応用(コネクション毎に認証を行う形態、署名、メッセージ認証など)を一枚のカードで可能。
- 3) ICカードのジャーナル機能をアプリケーションソフトウェアに開放することで個人に付随するデータにポータビリティを持たせ、セキュリティ以外に付加価値を追加。

このためカードリーダ・ライタは端末側に必須となるが、認証機能を独立させ、APIを通してその機能をアプリケーションに提供することで複数のアプリケーションで共通のスマートカードを利用可能としている。

2.4 認証方式

各エンティティ間の認証は全て相互認証を採用

している。相互認証が成立しない限りスマートカードはカード内の情報書き換えを認めない。

また再送攻撃を防ぐ目的から、サーバ側認証にはチャレンジ・レスポンス方式を、ユーザ側認証についてはタイムスタンプ方式を採用している。認証サーバでは自分の時刻とユーザから提示のあった時刻並びに、そのユーザが最後に認証成功した時刻と比較することで二重のチェックを行っている。時刻情報はあまねく公平に共有される情報であり、認証サーバはクライアント端末から時刻要求があった場合ならびに認証失敗した場合に正しいタイムスタンプを送付する。

本認証システムは認証アルゴリズムに非依存であり、現状では共通鍵暗号であるDESを実装している。閉じたグループ内で利用する形態であることと、スマートカードの現在の処理能力および経済性を考慮すれば共通鍵方式が実用的である。

2.5 構成

現在認証機能はトランスポート層の上位に位置しAPIを通してアプリケーションに提供される。試作した認証システムは以下に示すエンティティで構成されている。

認証サーバ(AUC)

カード(ユーザ)に関する情報を一元的に管理する。この情報を元にカード認証を行い、本人証明書をカードに発行する。

分散認証サーバ(AUS)

認証サーバが発行した証明書を鑑定し、正当であればアプリケーションサーバの利用許可書を発行する。

アプリケーションサーバ(APS)

利用許可書を鑑定し、正当であれば要求サービスをクライアント端末に対して提供する。

クライアント端末(CLNT)

認証処理のためにスマートカードと各サーバ間の仲介をし、アプリケーションサービスをユーザに伝達する機能を持つ。

スマートカード(CARD)

暗証コードによるユーザ認証手段を有し、ユーザ固有の認証情報を安全に格納するとともに、ネットワークを介した各サーバ間との相互認証プロセスをカード内で閉じて実行する。

なお現状ではサーバは全てSun OS 4.1.3上で、クライアントはSun OS 4.1.3ならびにWindows3.1上で作成している。

2.6 前提条件

以下の前提のもとで運用されるものとする。

- 1) スマートカードは物理的な攻撃に対して安全である。
- 2) サーバは安全に運用されている。
- 3) サーバおよびクライアント端末は時刻同期している。

2.7 動作

図1に認証の流れを示す。認証は次の4つのフェーズに分けられる。

- (a) PIN(Personal Identification Number) 認証
- (b) 本人証明書申請・取得
- (c) 許可証申請・取得
- (d) アプリケーションサーバの利用

(a) PIN認証

現在アクセスしようとしているユーザが正当なスマートカードの所有者であることを確認するためにPINによる認証を行う。PINはスマートカードとユーザ間で共有される。PINがクライアント端末で盗まれたとしても、スマートカードそのものもしくはそこに格納された認証鍵が盗まれない限り偽装できない。PINはスマートカード内で照合され、連続失敗回数によってカードロックすることができる。なお、スマートカードにキーパッドがあるものやカードリーダーにキーパッドがある場合にはPINがクライアント端末を経ずともPIN認証が可能である。

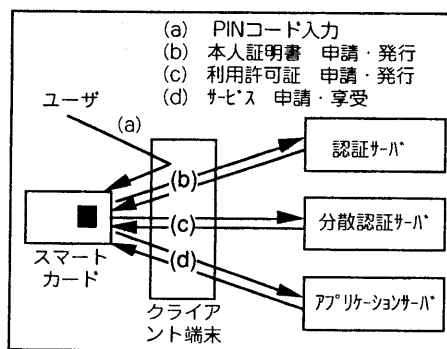


図1 認証の流れ

(b) 本人証明書 申請・発行

図2に認証プロトコルを示す。図中に用いられている記号は表1で与えている。

- (b1) クライアント端末においてタイムスタンプTSclntを生成しカードに送る。
- (b2) カードはタイムスタンプを暗号化したREScardならびにサーバ認証用の乱数RNDcardを生成し、カード番号IDcardとともにクライアント端末へ送る。
- (b3) クライアント端末ではカードからのメッセージに、自分のアドレスIPclntを加えて認証サーバへ通知する。
- (b4) 認証サーバは、カード番号が登録されていることと、ブラックリストに登録されていないことを確認し、クライアントのタイムスタンプをサーバ側タイムスタンプTSaucと比較する。また図には示していないが、そのカードが最後に認証成功した時刻(last-auth-time)とも比較し、認証が成功する毎にlast-time-authを更新している。次にカードと分散認証サーバ間のセッション鍵Kcard-ausを作成し、証明書CERTを作成する。証明書は分散認証サーバ用とカード用の2通を作成する。また相互認証用のレスポンスRESaucを生成し、これらをクライアント端末へ返す。
- (b5) クライアント端末は認証サーバから受け取ったメッセージをカードへ転送する。このメッセージはクライアント端末では解読できない。
- (b6) カードでは、サーバ認証を行い正当であればメッセージの解読を行う。カード用に発行された証明書は解読できて、分散認証サーバ間とのセッション鍵Kcard-ausは復号後に読み出し禁止にする。分散認証サーバ用の証明書は復号した後に格納する(鑑定はできない)。最後にクライアント端末に対して認証サーバのタイムスタンプTSaucを沿えてクライアント端末に相互認証結果を通知する。

なお証明書は、一定期間内であれば再利用可能であり、既に発行された証明書が有効であればこのフェーズは省略できる。これによって認証サーバの負荷を軽減している。

表1 記号の意味

エンティティ	
CARD	スマートカード
CLNT	クライアント端末
AUS	分散認証サーバ
AUC	認証サーバ
識別子	
IDcard	カード番号
IPxxx	エンティティxxxのIPアドレス
鍵	
Kxxx	エンティティxxxの秘密鍵
Kxxx-yyy	エンティティxxxとyyy間のセッション鍵
認証要素	
RNDxxx	エンティティxxxが生成するチャレンジ(乱数)
RESxxx	エンティティxxxが生成するレスポンス
TSxxx	エンティティxxxが生成するタイムスタンプ
REPxxx	エンティティxxxが生成する返答メッセージ
CERT	本人証明書
LIC	サービス利用許可書
操作など	
{ppp}_kkk	平文メッセージpppを鍵kkkで暗号化
ccc-ppp	暗号文メッセージcccを復号してpppを取り出す
データベース	
ZZZlist = {a}	aの要素を格納したデータベースZZZ
判断	
a = b ?	aとbは等しいか?
a c d d d ?	aはデータベースdddに含まれるか?

(c) 利用許可証 申請・発行

図2および表1にその手続を示す。本人証明書の発行と同様であるが、カード番号IDcardではなく分散認証サーバ用本人証明書(CERT)_Kausで識別し、認証鍵Kuserではなく本人証明書に記入されたセッション鍵Kcard-ausで相互認証が行われる。最終的にアプリケーションサーバで一度きり有効な利用許可証LICがユーザに対して発行される。また、カードで復号されるセッション鍵Kcard-apsはクライアント端末へ引き渡されるので、認証終了後にアプリケーション間の通信秘匿などに使用される。

(d) アプリケーションサーバ利用

クライアント端末では獲得したサーバ利用許可書を提示することでアプリケーションサーバからサービスを提供してもらう。これについては次節でその応用例とともに述べる。

3 WWWシステムへの応用

3.1 概要

前節で紹介した認証システムの応用としてWWWサーバ/ブラウザと連携した認証ソフトウェアを試作した。これはアクセスフリーの情報については認証することなくWWWサーバからハイパーテキストをブラウズでき、アクセス制限のある情報についてはスマートカードによるユーザ認証を経た後でブラウズすることができるものである。

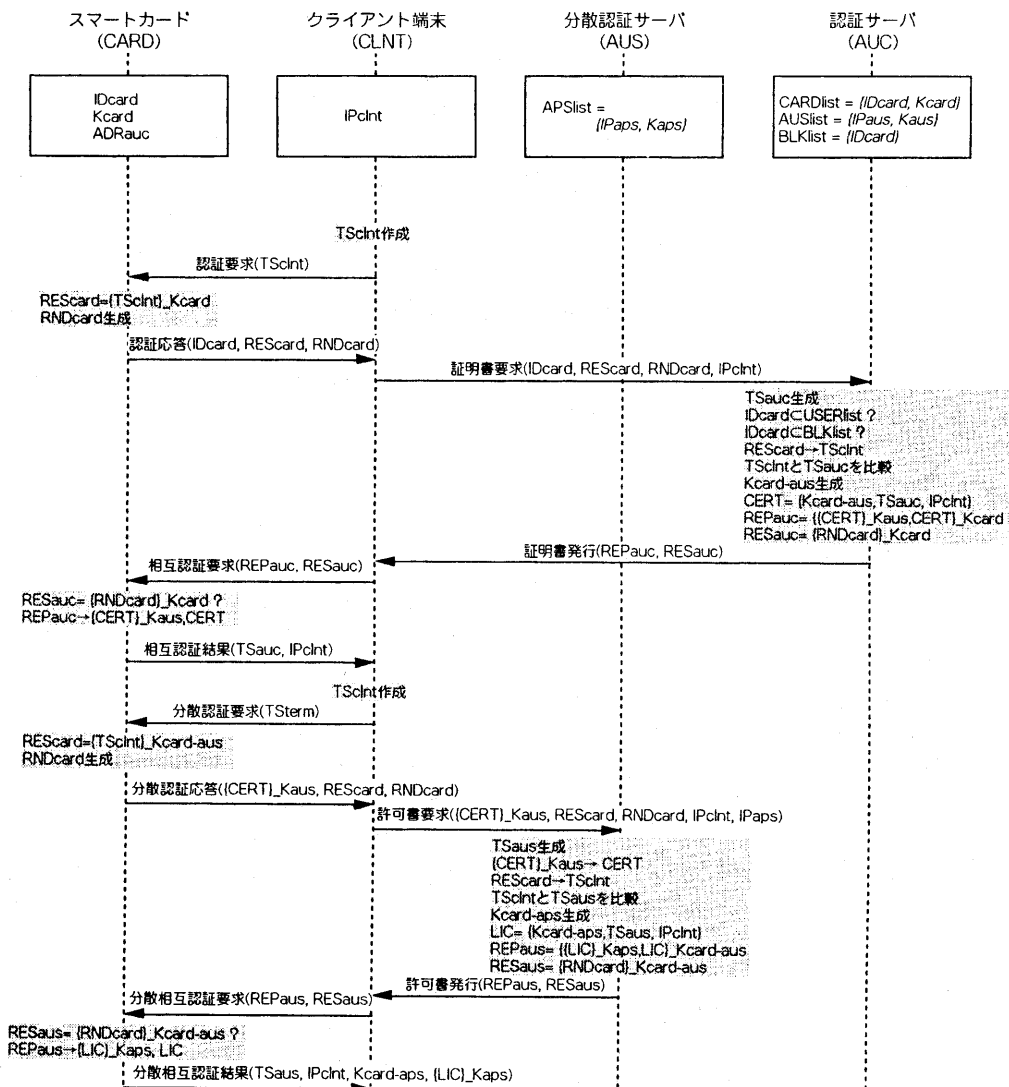


図 2 認証プロトコル

3.2 構成

アプリケーションサーバにはhttpdとファイル転送デーモン(以後ftdと称する)を、クライアント端末側にはアプリケーションとしてWWWブラウザ(Mosaicなど)とファイル転送クライアントデーモン(以後ftcdと称する)が存在している。

WWWブラウザでは、ftcdは一つのヘルパーアプリケーションとして登録、認識されている。また、サーバ側ではCGIを介してhttpdとftdが連携をとっている。なおクライアント端末はSun上で作成している。

3.3 動作

図3に沿って動作の流れを述べる。ここでユーザがアクセス制限情報を要求すると、httpdはftdのパス(例えば、./cgi-bin/ftd)が記述されたhtmlのsubmit formをクライアント端末に送る。ユーザに対しては要求情報がアクセス制限されている旨のメッセージを発する。

(1) アクセス制限情報の要求

クライアント端末がサーバに対してsubmit formのメッセージを返送すると、サーバ側ではhttpdがftdに対して、クライアント端末のアドレ

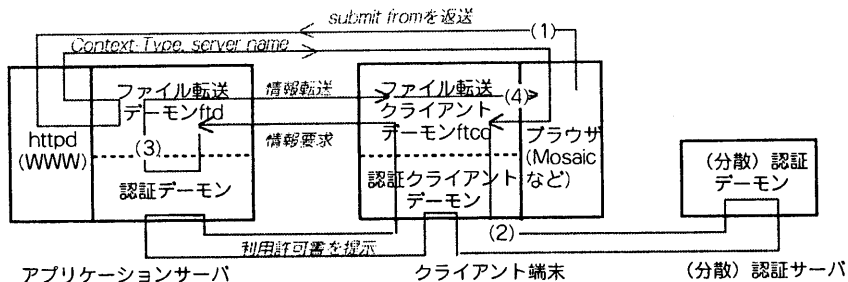


図3 応用システムの流れ

スと要求情報番号(インデックス)を通知する.ftdではこれらの情報と要求のあった時間を登録する。クライアント端末に対しては、httpdを介して、Content-Typeとアプリケーションサーバ名を送信する。Content-Typeにはアプリケーションとしてftcdが指定されている。ブラウザではこれに基づいて、ftcdをキックし、アプリケーションサーバ名を引き渡す。

(2) ユーザ認証

2節で紹介した、スマートカードによる認証プロセスが実行される。図3では認証サーバを統一しているが、実際には2種類(AUCとAUS)存在し、最終的にクライアント端末はサービスの利用許可書を得、アプリケーションサーバへ提示する。利用許可証にはクライアント端末のアドレスも記入されている。以上の手順が終了した時点でセッション鍵がアプリケーションサーバとクライアント端末間で共有される。

(3) 許可証の鑑定とファイルの転送

アプリケーションサーバでは、ftdがAPIを通して認証デーモンを呼び出し、認証結果、セッション鍵ならびに、許可証に記述されているクライアント端末のアドレスを取得する。ftdでは認証が成功している場合に、次の点を確認する。

- (a) そのクライアント端末アドレスは(1)において情報の要求を登録をしているか
- (b) その登録時刻が一定時間以内であるか

これらについて確認されたとき、要求された情報の転送を行う。セッション鍵Kcard-apsを用いて暗号化して情報を転送している。

(4) ファイル転送とそのブラウザ

最後にクライアント端末では、別画面のブラウザをキックし、転送されたファイルにそのポイン

タをはるることにより、転送情報をユーザに提示する。

4 おわりに

本稿では、スマートカードを用いた認証システムの開発とそのインターネット上への応用としてWWWシステムへの組み込んだ例について報告した。本認証システムは相互認証を採用するとともにセキュリティに関わる認証部分をカード内で行うことにより、ユーザがアクセスする端末に依存しない安全性を確保している。またWWWシステムへの認証方式の実装では、httpdおよびブラウザソフトを改修することなく、ユーザに対してアクセス制限付き情報転送サービスを実現できた。

今後は署名などの機能を視野にいれ公開鍵暗号の実装について検討する予定である。最後に日頃ご指導いただくKDD研究所浦野所長に感謝いたします。

参考文献

- [1] 鈴木, 大橋, "ICカードを利用した分散処理型認証システム", 電子情報通信学会IN研究会, SSE94-108, September 1994.
- [2] 鈴木, 大橋, "スマートカードを用いたクライアント/サーバ型認証システムの開発", 情報理論とその応用学会シンポジウム, F13-6, 広島, December 1994.
- [3] 山口英 訳, "UNIXセキュリティ", アスキー出版局, 1993.