TRN：耐タンパ性ネットワーク

倉光君郎[†] 村上直[††] 坂村健[†]

ネットワーク上で配信されたコンテンツは一般に無制限に複製され増えていく．これは，インターネット上で電子商品を販売する業者の大きな悩みである．本論文は，電子商品は自由に流通できるが，けっして複製することができない環境として,「耐タンパ性ネットワーク（TRN）」を提案する．TRNは，耐タンパデバイス間の複製不可能転送を保証するため，2つのセキュリティ要素を備えている．1つは，オンライン認証局を必要としない公開鍵方式の相互認証機構である．もう1つは，悪意のある転送中断に対して，電子商品が2つに増えたり，消滅したりしない，トランザクション機能である．我々は，チケットの電子流通実験のため，TRNのプロトタイプ実装を行った．本論文では，TRNの設計と実験結果を報告する．

TRN: Tamper Resistance Network

Kimio Kuramitsu,[†] Tadashi Murakami[††] and Ken Sakamura[†]

Electronic products once distributed through networks increase by unlimited duplication. Illegal copies always disturb merchants selling electronic products on the Internet. This paper proposes “Tamper Resistance Network” (TRN) providing the distribution environments where we can exchange electronic products freely, but cannot duplicate them. TRN supports two security aspects for ensuring unduplicatable transfer between tamper-proof devices (TPDs). One is an applied public-key mechanism that enables two TPDs (without online certificate authority) to authenticate trust with each other, and to open secure sessions. Another is an atomicity that moves electronic products without possibilities of creating and destroying them. This paper also addresses the prototyped TRN implementation for the distribution experimentation of electronic tickets.

1. はじめに

今日のインターネットコマースは，音楽データやデジタルチケットなど電子商品のネット流通の可能性を広げている．しかし同時に不法な複製は，コンテンツ提供者の頭痛のタネでもある．最近では，スマートカードやコピー防止機能付メモリカードなど，特殊なハードウェアの保護機能を用いて，流通した電子商品の「価値」を守る方法が一般的になりつつある[2),5)]．

耐タンパ性デバイス（TPD: Tamper Proof Device）は，不法な複製や不正更新から蓄積されたデータを守るようにハードウェア設計されている．しかし，データの流通性の観点からみれば，デバイス上の耐タンパ性のみでは不十分である．本研究の目的は，耐タンパ性をTPDで結ばれたネットワーク環境に拡張し，流通する電子商品の価値の一貫性を保証することである．我々は，そのような環境を「耐タンパ性ネットワーク（TRN: Tamper Resistance Network）」と名付けた．

我々が本論文で想定するTRNアプリケーションの1つは,「電子チケットシステム（Electronic Ticketing）」である．チケットは本質的に購入者が第三者に譲渡することを認めている．つまり，最終的な所有者がそのチケットを利用することができる．このとき，TRNは電子化されたチケットをインターネット上の発券サービスから安全にダウンロードすることを可能にするだけでなく，TPDどうし間で自由に交換することを可能にする．そのうえ，TRN上で流通するチケット数の一貫性は保証される．

TRNを実現するためには，2種類のセキュリティ（通信セキュリティとトランザクションセキュリティ）が必要になる．

1. **認証通信（Authentication）：**各TPDは，暗号セッションを開始する前に，通信相手の信頼性を自律的に検証できなければならない．
2. **原子性（Atomicity）：**電子商品の転送は，たと

† 東京大学大学院情報学環
Interfaculty Initiative in Information Studies, The University of Tokyo
†† マセマテック株式会社
MathemaTec Corporation

え悪意のある意図的な中断によっても，2つに増えてはならないし，消えてもならない．

我々は，通信システムを簡略化するため，2者間通信環境を前提として，TRNを実装するためのプロトコルスキームを設計する．設計されたスキームは，DH公開鍵認証方式と2相コミットプロトコルが応用され，オンライン上の信頼できる第三者の助けなしに，電子商品の転送を完結することが可能である．加えて，電子チケット実験プロジェクトにおけるTRNのプロトタイプ実装を報告し，2つのTPD間において直接相互認証し，セキュアセッションを開き，チケットをアトミックに転送可能であることを示す．

本論文の構成は以下のとおりである．2章では，TRNの概観を述べ，その要求をまとめる．3章では相互認証とアトミック転送の点で，プロトコルスキームを設計する．4章では，実験プロトタイプシステムの実装を述べる．5章では，セキュリティやアトミシティを評価する．6章では本論文を総括する．

2. 電子転送可能な商品

2.1 耐タンパ性デバイス

耐タンパ性デバイス（TPD）は，蓄積されたデータを不正な開閉や改ざんから保護する特別のハードウェアである．メモリとその操作ロジックは，不可分に設計されている．デバイス外部からのアクセスは，完全に制御され，必要に応じて暗号化されている．**図1**は，ワンチップマイコン型のTPDの内部構成を図示している．名前が示すとおり，CPUや暗号コプロセッサ，RAM，不揮発メモリ，通信チャネルなどの部品が単一チップ上に搭載されている．

スマートカードは，チップがプラスティックケースに埋め込まれた最も典型的なTPDである[5]．しかし，現在の技術水準ではスマートカードの処理能力と記憶容量は限られている．たとえば，最新の非接触カードの場合，8〜16ビットCPU，16〜32Kバイト ROM，512バイト RAM，データ記録用に4〜16Kバイト EEPROM程度である．チップの性能は，大幅に向上しているが，コンピューティング資源の限界に注意を払う必要がある．

しかしスマートカードは必ずしも唯一のTPDの形状ではない．プラスティックカードの代わりに，PDAや携帯電話の中に，セキュアチップを埋め込んだ個人デバイスも想定される（図1）．我々は，この種のデバイスが提供する組み込みユーザインタフェースや通信機能によって，一般ユーザの使い勝手は大幅に向上できると期待している．

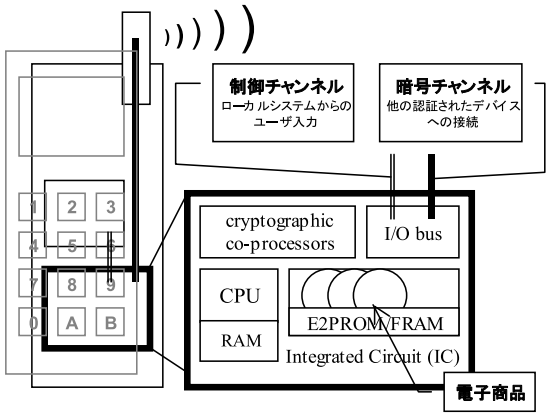

図1 耐タンパ性を備えたワンチップコンピュータ
Fig. 1 A single-chip microcomputer with tamper-proofness (TPD).

2.2 価値転送と関連研究

電子商品は，価値を持ったデジタルオブジェクトである．そのようなオブジェクトの移動は，一般に価値の2重化や破壊の危険性がある．そのため，電子商品を扱うシステムにとって，価値転送の制御は死活問題である．

多くのシステムでは，オンライン転送を制限している．スマートカードは価値を電子的に蓄積するが，その移動は物理的にカードに交換するための応用例と見なすことができる．また最近，NTTデジタルチケットシステムでは，スマートカード上のチケットコンテンツを信頼できる第3者を中継して交換する方法を提案している[12]．しかし，そのような中継局アプローチはシステムを複雑にし，プライバシ侵害への疑念を持つ人も現れる．

耐タンパ性ネットワーク（TRN）の目的は，電子化された価値を安全かつ直接的に2つのTPD間で転送を行うことである．言い換えれば，TRNはTPD間のVPN（バーチャルプライベートネットワーク）[6),10)]であり，オープンネットワーク上で価値を転送するため，トンネルを作り，相互に認証し，暗号通信を行う．**図2**は，ネットワーク階層におけるTRN層を示したものである．TRN層より上において，価値トークンの一貫性は保証される．

TRNと従来のVPNの大きな違いは，TPDは本質的に受動デバイスであり，セッションを自ら作ることができない点である．TPDは，TRNセッションを開くために外部システムの助けを必要とするが，その外部システムを信用することはできない．スマートカードと外部システムの間のセキュア通信プロトコルは数多く提案されている（代表例に，Mondexの Value

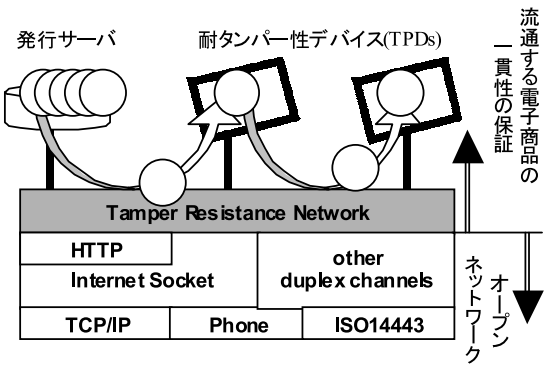

図 2 耐タンパ性ネットワークの概念的レイヤ
Fig. 2 A conceptual layer of Tamper Resistance Network.

Transfer Protocol[13]がある）．これらに対し，TRN の特徴は，対称的な関係にある TPD 間の通信の信頼性に着目し，特に外部システムの助けをなしに転送のアトミシティ[11]を保証する点である．不正に中断された通信においても価値が 2 重化されたり消えたりすることなく，自律的に転送を完了することを目標とする．

2.3 技術的な要求

TRN は，2 種類のセキュリティ（コミュニケーションとトランザクション）が必要となる．コミュニケーションでは，単なる通信の暗号化以外に，特に次のセキュリティ機能が必要となる．

- **通信相手の認証：**TPD は，電子商品を複数の TPD へ同時に転送しないようにしなければならない．通信相手の認証においては，通信相手の信頼性の検証だけでなく，単一性の保証も必要である．
- **再生攻撃の検出：**ある悪意のあるユーザは転送中の暗号化された電子チケットを複製して，システムを欺こうとするだろう．暗号通信では，そのような再生攻撃を検出できなければならない．

TRN は，加えて，外部トランザクション調停者なしで転送の ACID 性を保証することを目標としている．我々は，転送トランザクションの ACID 性を次のように定義している．

- **Atomicity：**価値転送は，閉じた 2 つの TPD 間の all-or-nothing 処理である．中断された送信は，転送済みか転送前のどちらの状態で完了できなければならない．
- **Consistency：**価値転送は，信頼できない環境で行われるため，中断された転送の再開タイミングは予想することができない．転送処理中は，価値が 2 重化したり，完全に消えたりしないように一貫性を保証できなければならない．
- **Isolation：**TPD は，受信した電子価値を別の新しいトランザクションとして，次のデバイスに転送できなければならない．
- **Durability：**TPD は，中断したトランザクションを回復し，再同期するために必要なロギング情報を記録する必要がある．

最後に，TRN スタックは計算機資源の制限された TPD 上に搭載可能なコンパクトデザインも求められている．

表 1 本論文におけるプロトコル表記
Table 1 Notation used for protocols throughout the paper.

A → B: m	あるデバイス A から他のデバイス B に対して，メッセージ m を送る．
m1, m2	メッセージ m1 と m2 の連結
k(m) sk(m) pk(m)	対称鍵 k，秘密鍵 sk，公開鍵 pk で暗号化されたメッセージ
$[m]_A$	デバイス A によるメッセージ m の電子署名
h(m)	メッセージ m のハッシュ値

3. プロトコルスキームの設計

TRN プロトコルは，2 つの段階（認証とトランザクション）から構成される．我々は，全 2 重チャネルの利用を前提として，プロトコルスキームを設計した．**表 1** は，本論文を通して用いる表記法をまとめてある．

3.1 相互認証

一般的に，スマートカードは秘密情報を安全に保持するデバイスであるため，チャレンジ=レスポンス型認証で実用的に安全とされる[5]．しかし，我々は将来の TRN の大規模な運用に備えて，PKI（Public Key Infrastructure[1]）の基盤を導入する．

3.1.1 デバイスの定義

まず，我々は耐タンパデバイスを公開/秘密パラメータの集合として定義する（デバイス外に完全に秘匿される秘密値は，下線付きで表記する）．

ID_X を耐タンパデバイス $\mathrm{TPD_X}$ の識別子，pk_X と $\underline{sk}_X$ を公開鍵/秘密鍵のペアとする．CA 局は，それらに対して，デバイスの信頼性証明書として電子署名 $[ID_X, pk_X]_{\mathrm{CA}}$ を発行する．さらに，耐タンパデバイスは，認証時オンライン CA 局を参照できないため，証明書を検証するため，CA 局の公開鍵 $\underline{pk}_{CA}$ が必要である．数値 $\underline{a}$ は，認証されたセッション用の対称鍵を生成するための秘密シードである．

$\mathrm{TPD_X}$ は，したがって，初期状態において次のパラメータを保持している．

$$X = \left\{ ID_X, pk_X, [ID_X, pk_X]_{\mathrm{CA}}, \underline{pk}_{CA}, \underline{sk}_X, \underline{a} \right\}$$

3.1.2 相互認証とセキュアチャンネルの生成

お互いに認証を始めた耐タンパデバイスは，対称的に認証プロセスを進める必要がある．もしどちらか一方が信頼できないと感じたら，認証中のセッションはただちに閉じられる．ここでは，$\mathrm{TPD_A}$ と $\mathrm{TPD_B}$ 間の相互認証を定義する．

Step1：デバイス証明書の交換と検証 耐タンパデバイスは外部から初期化されたとき，デバイス証明書を通信相手に送出する．

A → B: $ID_A, pk_A, [ID_A, k_A]_{\mathrm{CA}}$

B → A: $ID_B, pk_B, [ID_B, k_B]_{\mathrm{CA}}$

証明書を受信した通信相手は，$\mathrm{pk_{CA}}$ を用いて，証明書を検証する．検証結果が正しければ，受信したデバイス証明書は CA 局が発行したものと信じることができる☆．

Step2：乱数を生成し，電子署名を付けて交換 $\mathrm{TPD_A}$ と $\mathrm{TPD_B}$ は，それぞれ乱数シード（$\underline{\mathrm{rnd}}_{\mathrm{A}}$ と $\underline{\mathrm{rnd}}_{\mathrm{B}}$）を生成し，それを受け取った相手の公開鍵で暗号化する．さらに，その暗号文は，中間攻撃（*man-in-the-middle-attack*）を避けるため，それぞれのデバイスの秘密鍵（$\underline{sk}_A$ and $\underline{sk}_B$）を用いて電子署名を発行する．

A → B: $\mathrm{pk_B}(\underline{\mathrm{rnd}}_{\mathrm{A}}), [\mathrm{pk_B}(\underline{\mathrm{rnd}}_{\mathrm{A}})]_{\mathrm{A}}$

B → A: $\mathrm{pk_A}(\underline{\mathrm{rnd}}_{\mathrm{B}}), [\mathrm{pk_A}(\underline{\mathrm{rnd}}_{\mathrm{B}})]_{\mathrm{B}}$

Step3：対称鍵の生成とお互いの挨拶 $\mathrm{TPD_A}$ は，暗号化された乱数 $\mathrm{pk_A}(\underline{\mathrm{rnd}}_{\mathrm{B}})$ を sk_A を用いて復号化し，署名 $[\mathrm{pk_A}(\underline{\mathrm{rnd}}_{\mathrm{B}})]_{\mathrm{B}}$ を pk_B を用いて検証する．このプロセスが正しく完了すれば，$\mathrm{TPD_A}$ は $\mathrm{rnd_B}$ より，$\mathrm{TPD_B}$ と共通の対称鍵を生成できる．生成アルゴリズムは，次のとおりである．

A: $\mathrm{h}(\underline{a}, \underline{\mathrm{rnd}}_{\mathrm{B}}) => \underline{\mathrm{k}}_{\mathrm{B}}$

B: $\mathrm{h}(\underline{a}, \underline{\mathrm{rnd}}_{\mathrm{B}}) => \underline{\mathrm{k}}_{\mathrm{B}}$

$\mathrm{TPD_B}$ も受信した $\mathrm{rnd_A}$ が信頼できるとき，同様に (i.e., $\mathrm{h}(\underline{a}, \underline{\mathrm{rnd}}_{\mathrm{A}}) => \mathrm{k_A}$) 鍵を共有できる．

最終的に，$\mathrm{TPD_A}$ と $\mathrm{TPD_B}$ は，挨拶文を交換し，セキュアセッションを確認する．

A → B: $\mathrm{k_A}$ (“hello”)

B → A: $\mathrm{k_B}$ (“hello”)

セキュアセッションが開いた時点で，$\mathrm{TPD_A}$ と $\mathrm{TPD_B}$ はそれぞれ以下のパラメータを保持している．

$$A = \{ID_A, pk_A, [ID_A, pk_A]_{\mathrm{CA}}, \underline{pk}_{CA}, \underline{sk}_A, \underline{a}, \mathrm{ID_B}, \underline{\mathrm{k}}_{\mathrm{A}}, \underline{\mathrm{k}}_{\mathrm{B}}\}$$

☆ この段階では，$\mathrm{TPD_A}$ は通信相手が $\mathrm{TPD_B}$ であると信用できない．なぜなら，$[ID_B, k_B]_{\mathrm{CA}}$ は公開情報でだれでもなりすますことができるからである．

表 2 商品表記の例
Table 2 An example of product notations.

商品表記	説明
$p\,(O, enabled, ANY)$	完全な商品
$p\,(O, enabled, NO)$	転送不可能な商品
$p\,(O, disabled, ID_B)$	$\mathrm{TPD_B}$ のみ転送可能な商品
$p\,(NULL)$	メモリ上から消去された商品

$$B = \{ID_B, pk_B, [ID_B, pk_B]_{\mathrm{CA}} \underline{pk}_{CA}, \underline{sk}_B, \underline{a}, \mathrm{ID_A}, \underline{\mathrm{k}}_{\mathrm{A}}, \underline{\mathrm{k}}_{\mathrm{B}}\}$$

通信相手の識別子と対称鍵は，次節の転送トランザクションで利用される．

3.2 転送トランザクション

もし電子商品が単なる単一のオブジェクトとして転送されれば，送信者と受信者は ACK のための ACK を延々と繰り返さなければならない．我々は，まず電子転送可能商品の概念的なモデル化から始めた．

3.2.1 電子商品のモデル定義

電子商品は，デジタルオブジェクトである．我々は，そのオブジェクトを次のような 3 つの概念的な要素に分割した．

1. **コンテンツ（Content）** [表記 $O/oid/NULL$]：O は，完全なコンテンツ情報を含んだオブジェクトを表す．その中には，オブジェクト識別子 oid も含まれている（i.e., $oid \subset O$）．$NULL$ は，O と oid のメモリ上の消去を明示的に表現している．
2. **価値（Value）** [表記 $enabled/disabled$]：コンテンツ (O) の所有者が，電子商品としてそれを利用できるかどうか表している．
3. **移動権（Right of Transfer）** [表記 $ANY/ID_X/NO$]：表記 ANY/NO は，ユーザが O を任意の耐タンパデバイスに移動できるか，もしくはどのデバイスにも移動できないかを表している．ID_X は，指定されたデバイス $\mathrm{TPD_X}$ に限定して，移動可能であることを示している．

電子商品の表記： 電子転送可能商品 p は，3 要素の状態の組み $p\,(content, value, right\ of\ transfer)$ として表記される．唯一の例外は，消去された商品であり，$p\,(NULL)$ のように表記される．**表 2** は，本論文でしばしば用いられる商品表記をまとめたものである．

3.2.2 転送トランザクションの定義

転送トランザクションは，$\mathrm{TPD_A}$（送信者）と $\mathrm{TPD_B}$（受信者）の間における電子商品 p の状態遷移として定義する．**表 3** は，初期状態 $T\,(init)$，転送成功 $T\,(succeed)$，転送失敗 $T\,(aborted)$ を定義している．

3.2.3 2 相コミットプロトコルの適用

2 相コミットプロトコル（2PC: two-phase commit

protocol）は，分散システムのトランザクション状態の同期をとる基本的枠組みである[3]．我々は，2PCを2者間の転送トランザクション T の状態遷移に適用した．M={“copy”, “copied”, “commit”, “committed”, “true”, “false”}をプロトコルメッセージの集合とすると，$T(succeed)$ は，次のように定義される．

A：p を使用不可能にし，送り先を ID_Bに固定する；
$p(O, enabled, ANY) \rightarrow p(O, disabled, ID_B)$
A → B: sk_A(“copy”, O)
B：複製する；$p(O, enable, NO)$
B → A: sk_B(“copied”, true)
A：B上の oid に対して，移動権をコミットする
A → B: sk_A(“commit”, oid)
B → A: sk_B(“committed”)

もし TPD_B（受信者）がメモリ不足や所有者の拒否により複製した O を蓄積できなかったら，TPD_B は $T(failed)$ のケースとして，sk_B(copied, false)を返す．**図3**は $T(succeed)$ と $T(failed)$ における完全なシーケンスを表している．

4. プロトタイプ実装

我々は，民間企業と協力して，TRN上の電子チケットの流通実験を行った．本章では，実験で用いたプロトタイプ実装について述べる．

表3 転送 p における状態遷移
Table 3 The status transition in transfer p.

	TPD_A（送信者）	TPD_B（受信者）
$T(init)$	$p(O, enabled, ANY)$	$p(NULL)$
$T(succeed)$	$p(NULL)$	$p(O, enabled, ANY)$
$T(failed)$	$p(O, enabled, ANY)$	$p(NULL)$

4.1 実験環境

実験環境は，次の想定シナリオを含んでいた．(1) インターネット上でチケットを購入したユーザは，直接，発券サーバからチケット端末（TPD）に電子チケットをダウンロードすることができる，(2) ユーザは，電子チケットをチケット端末間で自由に交換することができる，(3) 最終的に，入場ゲートシステムでチケットを検証する．**図4**は，実験システムの全体の流れを示している．

4.2 通信系

我々は，チケット発券サービスをWeb/Servletサーバ上に構築した．サーバは，その信頼性を示すため，仮想的な耐タンパデバイスとしてTRNプロトコルが実装されている．具体的には，発券Servletは，Switching-Protocol (101) 機構によって，HTTP/1.1セッションからTRN認証モードに切り替えている．

PCは，他方，チケット端末とR/W装置を通して通信する．R/W装置とチケット端末の間は，マイクロ波（ISO14443）を用いた非接触通信である．我々は，低レベル物理層の上に，TRNセッションのため，全2重化するトランスポート層を実装した．

発券サーバとチケット端末（TPD）は，基本的に受動デバイスである．そのため，利用者は，PCを用いて，TRNセッションを初期化する．利用者がチケットを購入する場合，まずHTTPリクエスト（“Protocol-Upgrade: TRN”）をサーバに送り，サーバセッションをTRN認証モードにスイッチする．同時に，チケット端末をポーリングし，R/W側のTRNセッションも初期化する．PCは，中継システムとして，認証メッセージとそれに続く暗号文を交換する．PCは，受動的なTPDを起動し中継するだけである．サーバとチケット端末は，けっしてPCシステムを信頼すること

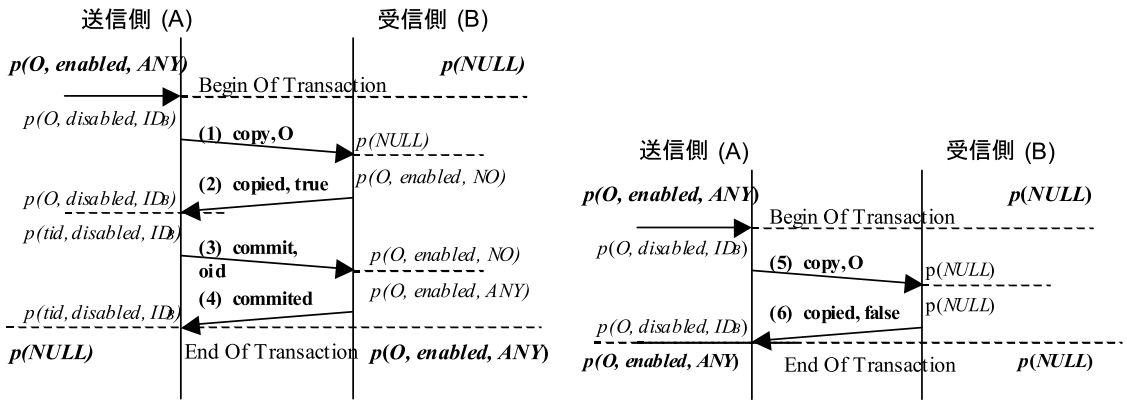

図3 2相コミットプロトコル
Fig. 3 2-phase commit protocol.

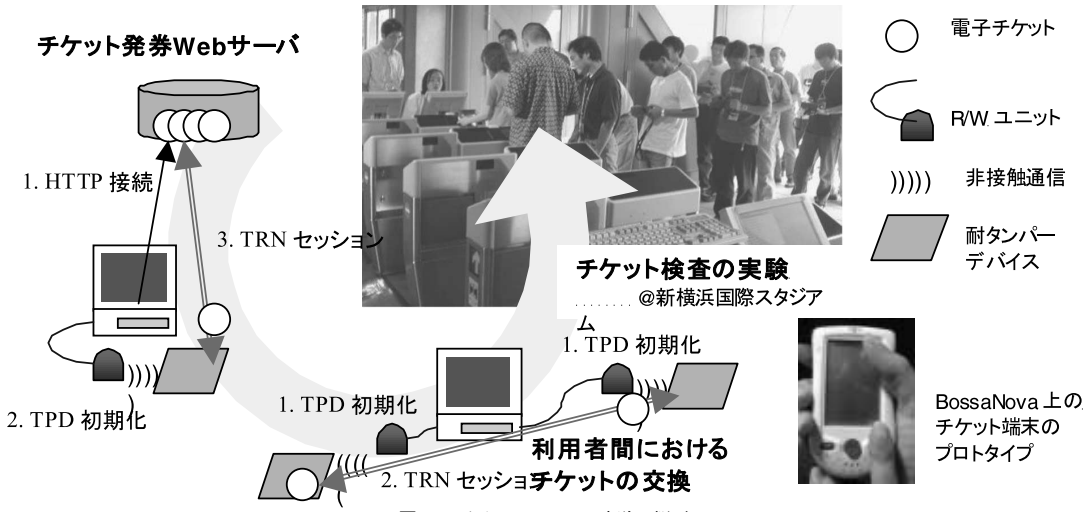

図 4 デジタルチケット実験の概要
Fig. 4 The overview of experimental environment for digital ticketing.

はない．同様に，チケット交換の場合，利用者は PC を用いて，2 つのチケット端末を初期化する☆．

4.3 チケット端末

我々は，まずはじめに PFU BossaNova[9] を利用してチケット端末のプロトタイプを実装した．BossaNova は，メモリに対する適切な耐タンパ性は備えていなかったが，JDK1.1 ベースで開発可能なタッチパネル式の GUI と IrDA 通信機能を備えていた．我々は，RSA 公開鍵方式の認証システムと DES 対称鍵方式の暗号通信路を実装した☆☆．

初期プロトタイプに続いて，我々はより本格的なチケット端末を開発した．それは，楕円曲線暗号エンジンと ISO14443 非接触通信機能を備えた専用耐タンパ性セキュリティボックスを利用している（このセキュリティボックスは，本来は，非接触型スマートカードの開発用エミュレータとして製作されたものである）．チケット端末や発券サーバは，シリアルを経由してセキュリティボックスの機能を利用して，TRN セッションの生成や通信を行った．

4.4 データ・フォーマット

我々は，電子チケットの表現形式として，μPCO フォーマットを利用した．μPCO は，PCO 言語[7),8)] のコンパクト版であり，スマートカード上の EDI 言語として再設計されている．μPCO によるチケット 1 枚のサイズは，およそ 800 バイトであった．μPCO 言語は，TPD 上の μPCO コンテンツを操作するためのクエリー・レスポンスのフォーマットも定義している．**図 5** は，今回実装したフォーマットのレイアウトである．そのフォーマットには，4 バイトのクエリとレスポンスのペアを識別するシーケンス番号を記録するフィールドがある．

	1	Type of Message (Encryption Mode)
	2	Packet Number
block	1	Size of Block (1block = 8bytes)
	2	Type of Query/Response
	4	Sequence Number for Query
	2	Size of *Body*
	n	*Body*
	0~7	Null Padding
	4	CRC Checksum

図 5 実装された TRN メッセージのレイアウト
Fig. 5 Layout of message in implemented TRN.

4.5 ま と め

我々のプロトタイプシステムは，100 人を超える被験者によって，チケット購入から相互交換，スタジアムへの入場まで，テストされた．1 枚のチケットを購入や交換するために必要な時間は，およそ 4 秒程度であった．

5. 評 価

この章では，TRN のセキュリティと ACID 性について評価する．

☆ PC は，チケット端末をリモートで操作するインタフェースとして利用することができる．我々は，TRN 暗号メッセージと混在させながら，制御コードやイベントなど非 TRN メッセージを交換できるように実装した（図 6 参照）．

☆☆ BossaNova 上の TRN 実装は，Servlet 版チケット発券サーバの実装に継承された．

5.1 セキュリティ

TRN 認証は，よく知られた DH 鍵交換/公開鍵方式[4]を基礎としている．つまり，SSL の軽量版と見なすことができる．その安全性は，ここでは簡単にプロトコル中（3.1 節）に秘密情報が平文で現れないことを確認するのにとどめておく．ここでは，TRN 独自の特徴であるオフライン時における偽デバイス証明書攻撃と電子商品複製のための再生攻撃の安全性を議論する．

偽デバイス証明書：TRN 認証の特徴は，オンライン CA の仲介なしでデバイス証明書を検出するため，あらかじめ TPD 上に $\underline{pk_{CA}}$ がインストールされている点である．しかし，理論的には CA 局の秘密鍵（sk_{CA}）がなければ，第 3 者証明書を発行することができない．もし最悪の場合，攻撃者がある正当な $\mathrm{TPD_X}$ の公開情報である pk_X から sk_X を数学的に解析したら，偽 $\mathrm{TPD_x}$ を作る完全な情報を得たことになる．通常のオンライン CA がある場合は，$\mathrm{TPD_x}$ を無効リストに追加すればよいが，TRN の場合，特定のデバイスを無効にすることができない．TRN は，この弱点をカバーするため，共通鍵を生成するとき，秘密シード $\underline{a}$ を用いることになっている．$\underline{a}$ は，単なる秘密値なので，$\mathrm{sk_X}$ を破ったケースと異なって，数学的に解析することはできない．

再生攻撃：効率の面から，TRN は同一セッション内で複数個の電子商品を送信することを認めている．そのため，悪意のあるユーザは，暗号プロトコルを解読することなし，通信パケットの内容を推測できるため，TRN トランザクションの偽装が可能になる．たとえば，電子商品の転送完了後，攻撃者はセッションを盗み，記録した転送プロトコル文を再送することで，複製を試みる可能性がある．TRN 実装（4 章）では，メッセージ上にユニークシーケンス番号を導入している．番号は，セッション生成時に乱数で設定され，各メッセージ交換ごとに 1 増える．したがって，送信者も受信者も攻撃者の侵入を検出でき，偽メッセージを取り除くことができる．

5.2 ACID 性の証明

電子商品 p の TRN 転送において，p の遷移状態は有限である．**表 4** は，送信デバイス（$\mathrm{TPD_A}$）と受信デバイス（$\mathrm{TPD_B}$）における p の状態ペアをすべて示している．**図 6** は，TRN 転送における状態遷移図を表示している．ここでは，p の転送における ACID 性を証明する．

- **Consistency：**転送が中断した場合，p の状態は必ず 4 つの状態 (II)〜(V) の 1 つをとる．これらのどの状態において，$\mathrm{TPD_A}$ と $\mathrm{TPD_B}$ の両端において同時に p の価値（value）が有効になることはない．さらに，どの中断点でも，少なくとも片方の TPD 上に，完全なコンテンツ O が残っている．したがって，TRN は，転送中の電子商品に対して，価値を 2 重化することもコンテンツを完全に消すこともない．
- **Atomicity：**TRN では，送信者（$\mathrm{TPD_A}$）が中断されたトランザクションを再開するコーディネータの役を担う．$\mathrm{TPD_A}$ からみれば，中断状態 (II) と (III)，さらに (IV) と (V) は区別することはできない．したがって，$\mathrm{TPD_A}$ は次の 2 つの方法で再開する．$\mathrm{TPD_A}$ 上の状態 $p\,(O, disabled, ID_B)$（(II) か (III)）の場合，$\mathrm{TPD_A}$ はメッセージ（A → B: copy, O）を再送する．$\mathrm{TPD_A}$ 上の状態 $p\,(oid, disabled, ID_B)$（(IV) か (V)）の場合，$\mathrm{TPD_A}$ はメッセージ（A → B: commit, oid）を再送する．
- **Durability：**再開メッセージ（copy O と commit oid）は，$\mathrm{TPD_A}$ 上の p の状態（それぞれ (II) と (IV)）から生成される．通信相手は，p において，ID_B として記録されている．
- **Isolation：**(V) の状態において，TPD_B のユー

表 4 電子商品の各状態
Table 4 The status of product p.

	$\mathrm{TPD_A}$	$\mathrm{TPD_B}$
(I)	$p\,(O, enabled, ANY)$	$p\,(NULL)$
(II)	$p\,(O, disabled, ID_B)$	$p\,(NULL)$
(III)	$p\,(O, disabled, ID_B)$	$p\,(O, enabled, NO)$
(IV)	$p\,(old, disabled, ID_B)$	$p\,(O, enabled, NO)$
(V)	$p\,(old, disabled, ID_B)$	$p\,(O, enabled, ANY)$
(VI)	$p\,(NULL)$	$p\,(O, enabled, ANY)$

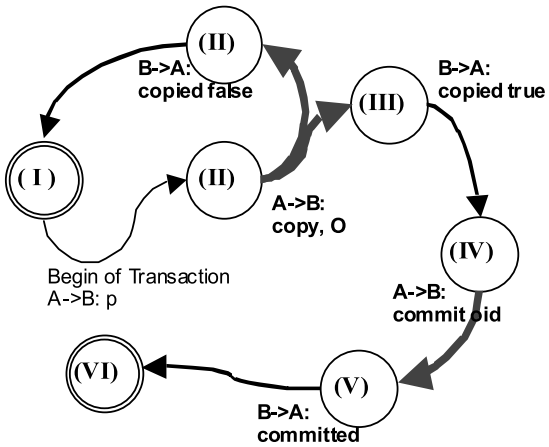

図 6 転送 p における状態遷移図
Fig. 6 The status transition diagram in transfer p.

ザは，他のデバイスに p を新たに送信することができる．なぜなら，committed メッセージは，$p(NULL)$ から生成することができるためである．

6. 結　　論

ネットワーク上で配信されたコンテンツは一般に無制限に複製され増えていく．電子転送製は，インターネット上で電子商品を販売する業者にとって，諸刃の剣である．耐タンパネットワーク（TRN）の貢献は，電子商品を自由に電子転送を可能にし，不正な複製から保護するネットワーク流通を提供することである．

TRN は，耐タンパデバイス間の複製不可能転送を保証するため，2 つのセキュリティ要素を備えている．1 つは，オンライン認証局を必要としない DH 公開鍵方式の相互認証機構である．もう 1 つは，悪意のある転送中断に対して，電子商品が 2 つに増えたり，消滅したりしない，トランザクション機能である．我々は，チケットの電子流通実験のため，TRN のプロトタイプ実装を行った．

TRN は，耐タンパ性デバイス（TPD）から他の TPD への信頼ある転送を保証するため，2 種類のセキュリティ要素を備えている．1 つは，2 つの TPD 間で自律的な VPN である．TRN は，DH/公開鍵方式を応用することで，オンライン CA なしで 2 つの TPD にお互いの信頼性を認証し，セキュアチャンネルを生成することができる．もう 1 つは，商品アトミシティである．TRN は，電子商品をコンテンツ，価値，移動権としてモデリングし，それに 2 相コミットプロトコルをあてはめることで，2 者間転送における ACID 性を保証する．

本論文では，TRN 上の電子チケット流通実験システムのプロトタイプを報告した．プロトタイプ TRN 実装は，TPD 間のチケット転送に十分な信頼性がもたらされることが示された．さらに，TRN のコンセプトは，電子転送可能な商品を扱うオンラインビジネスを立ち上げるのに有効であると結論づけられた．将来の研究の方向性は，実装アルゴリズムを含めセキュリティ評価，TRN 上の電子商品の追跡性，アトミックな交換性（fair-exchange），ビジネスモデルに応じた PKI デザインがある．

謝辞　本研究を進めるにあたりさまざまなご助力をいただいた越塚登先生，鵜坂智則先生，重定如彦氏，新堂克徳君，松田一君（東京大学）に深謝いたします．

参 考 文 献

1) Adams, C. and Lloyd, S.: *Understanding the Public-Key Infrastructure*, Macmillan Technology Series (1999).
2) Araki, S.: The Memory Stick, *IEEE Micro*, pp.40–46 (July-Aug. 2000).
3) Bernstein, P. and Newcomer, E.: *Principles of Transaction Processing*, Morgan Kaufmann Publishers (1997).
4) Diffie, W. and Hellman, M.: New Direction in Cryptography, *IEEE Trans. Information Theory*, IT-22, 6, pp.644–654 (1976).
5) Hansmann, U., Nicklous, M.S., Schack, T. and Seliger, F.: *Smart Card Application Development Using Java*, Springer (2000).
6) Kent, S. and Atkinson, R.: Security Architecture for the Internet Protocol, IETF RFC 2401 (Nov. 1998). available at http://www.rfc-editor.org/rfc/rfc2401.txt
7) Kuramitsu, K. and Sakamura, K.: Digiket: decentralized architecture for worldwide electronic market, *Proc. IEEE COMPSAC99* (Oct. 1999).
8) Kuramitsu, K. and Sakamura, K.: PCO: EC Content Description Language Supporting Distributed Schema across the Internet, *Journal of IPSJ*, Vol.41, No.1, pp.110–122 (2000).
9) PFU BossaNova Homepage (in Japanese), available at http://www.pfu.co.jp/BossaNova/
10) Townsley, W., et al.: Layer Two Tunneling Protocol (L2TP), IETF RFC 2661 (Aug. 1999). available at http://www.rfc-editor.org/rfc/rfc2661.txt
11) Tygar, J.D.: Atomicity in Electronic Commerce, *Proc. 5th Annual ACM Symposium on Principles of Distributed Computing* (May 1996).
12) Matsuyama, K. and Fujimura, K.: Distributed Digital-Ticket Management for Rights Trading System, *Proc. ACM EC'99* (Nov. 1999).
13) Mondex Electronic Cash. available at http://www.mondex.com/

(平成 12 年 12 月 1 日受付)
(平成 13 年 6 月 19 日採録)

倉光　君郎（正会員）

1972 年生．1996 年東京大学工学部卒業（機械情報工学）．1998 年東京大学大学院理学系研究科修了（情報科学）．2000 年同大学院博士課程中退．現在，東京大学大学院情報学環助手．電子商取引，半構造データ，インターネットマーケッティング技術に興味を持つ．IEEE，ACM 各会員．

村上　　直

1975 年生．1999 年京都大学工学部卒業（情報学科計算機科学コース）．2001 年東京大学大学院理学系研究科修了（情報科学）．現在，マセマテック株式会社にて数値計算に関する研究開発に従事．電子商取引，インターネット技術，マルチメディアに興味を持つ．

坂村　　健（正会員）

東京大学大学院情報学環教授．1984 年より TRON プロジェクトリーダーとして新しい概念に基づくコンピュータ体系の構築に精力を注ぐ．TRON は現在，携帯電話，デジタルビデオ，エンジン制御等，組み込みシステムの世界で最も使われている OS である．さらに最近はプロジェクトの最終目的である超機能分散システムの研究を進めている．