社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2004－UBI－3 (8)
2004／1／20

ユビキタス環境における権利回数制限の実装及び評価

小川博久[†1]

電子権利の安全な流通において，回数制限の実装は，重要な問題である．現在利用されている電子チケットでは，まとめ買いに対する利便性と安全性の問題が指摘されており，双方の要求を満たす技術が求められている．本稿では，(k,n)しきい値分散法及び，(k,L,n)しきい値分散法を用いた権利回数制限の構成例を示し，実装及び評価について報告する．

キーワード 秘密分散法，(k,n)しきい値法，(k,L,n)しきい値法，電子チケット，認証

Implementation and Evaluation of Limited Rights of Use in Ubiquitous Enviornment

Hirohisa OGAWA[†1]

Abstract.For a secure circulation of digital rights, it is an important issue to limit the number of use.It is often pointed out that digital tickets used nowadays have issues in convenience in regardsto bulk buying and security.This article makes an illustration of limited rights of use applying (k, n) threshold scheme and (k, L, n) threshold scheme, reports implementation and evaluation..

Keywords secret sharing scheme,(k,n)-threshold secret sharing scheme,(k,L,n)-threshold secret sharing scheme,E-ticket,authenticationHirohisa

1. はじめに

電子権利の安全な流通を考えた場合，発行された電子権利の権利情報と対価情報が変更されること無く交換される事が必要である．電子権利の発行者と所有者からなる権利該当者は，発行された電子権利が変更される事がないという前提で，安心して権利行使を行う事が出来る．

ここでいう電子権利とは，電子的な価値や権利を示し，対価情報となる金額か，期間や回数などを権利該当者同士が合意し定めた内容が内在している．この電子的な価値や権利が行使されるまで変更される事が無く処理される事は，電子商取引の安全性を支える重要な事である．

また，電子商取引推進協議会のモバイル電子チケット検討報告[1]では，モバイル環境における電子権利の流通に関する問題が3点指摘されている．

1. 携帯性について

権利情報は，通常，ネットワークに接続された状態で，扱われる．もし，権利所有者が，自らが所持している権利情報を確認する場合でもネットワークに接続し，外部に保管されている情報を閲覧する事になる．この場合，権利所有者は，権利所持者である属性認証を行うだけであり，実際の電子権利自体を所持携帯できていない．本来，権利所持者は，自らの電子権利を所持している事をどのような環境でも確認できるべきであり，自らが携帯している事が望ましい．権利所持者が電子権利を携帯していれば，ネットワーク接続されていない状態でも自らが所持している権利を確認する事ができ，安心する事ができる．この携帯性は，モバイル環境での問題として指摘されているが，同様にユビキタス環境においても言う事ができ，以下の問題点についても言える．

株式会社シーフォーテクノロジー
C4Technology,Inc.
oga@c4t.jp

2. 回数券の対応

モバイルチケット運用上の問題点として，回数券への対応が求められると報告されている．また，回数券の発行については，払戻し処理が発生することも考えられ，電子権利の返却を意味する．通常，電子権利の変容する使用サイクルは，金銭→電子権利→権利となるが，電子権利の返却では，金銭→電子権利→金銭となることが想定される．

報告では，旅行会社で予約購入5名で1枚発券，実際には4人乗車，1人分は不乗車証明発行を行い，1人分の権利期間を延長するか，電子権利の払い戻しが考えられる．このような場合は，分割可能とすべきだが携帯電話の場合はどうするか．と指摘されている．

3. まとめ買いと権利譲渡の対応

譲渡（転々流通）に関する利用者側の視点として，まとめ買いへの対応が求められている．

電子権利のまとめ買いとは，ある特定の人がチケットを複数枚買い求めて，（友人など）グループ・メンバに配布する「まとめ買い」行為は，配布するための仕組みが簡便でないと普及が難しいかもしれない．と記載されている．電子権利発行者として想定されるサービス事業者側の視点では，組織的な転売行為を防止したいが，電子権利所持者として想定される利用者側の視点としては，譲渡にコストがかからない事という相反する要求がある．

これらの問題に対して，本稿では，(k;n)しきい値分散法及び，(k,L,n)しきい値分散法によって権利情報及び，利用者のプライバシー情報を分散化し，分散化した情報の真正性を検証する事で，権利認証を行う権利認証スキーム[2]を実装し，権利の行使回数に対する制限機能の構成例を示し，実装及び評価について報告する．

第2節では，しきい値分散法を用いた簡易認証スキームによる権利認証の特徴を説明し，

第3節では，権利行使迄のプロトコル示し，回数制限の構成例及び，まとめ買いの概要を表す．また，まとめ買いに関して必要となる払い戻し行為である電子権利の返却プロトコルの構成例を示す．第4節では，回数制限及び，まとめ買いで想定される権利情報の譲渡のとプロトコルを示し，最後に回数制限を実装した評価結果を報告する．

2. しきい値秘密分散法を用いた簡易認証スキーム

2.1 しきい値秘密分散法

Shamir の(k,n)しきい値秘密分散法[2]は，有限体上のk-1次の多項式 $y=f_{(x)}$ を用いて秘密情報 S を $S=f_{(0)}$ として埋め込み，それを n 個の分散情報(Share) $a_J=f_{(j)},j=1,2,...n$，に分散符号化する方式である．このとき，k 個の Share,a_J,が集まると $f_{(x)}$ が一意に定まり，秘密情報 $S=f_{(0)}$ が求まる．しかし，k-1 個以下の a_J からでは，$f_{(0)}$ は全ての可能性があり，秘密情報 S について全く情報が得られない．

さらに，(k,L,n)しきい値法[3]では，しきい値に幅 L を持たせ．(k-L)個以下の a_J からは，S の情報が1ビットも漏れないが，(k-j)個，$0 \leqq j \leqq L$,の a_J では，秘密情報 S の曖昧さが全体の j/L だけ残るようにした方式である．この特性を持たせることにより，(k,L,n)しきい値法の Share のサイズは，(k.n)しきい値法に比べて 1/L にでき，符号化効率のよい秘密分散法となる．

2.2 しきい値秘密分散法を用いた簡易認証スキーム

既に提案しているしきい値秘密分散法を用いた簡易認証スキーム[4]を説明する．

このスキームでは，(k,n)しきい値秘密分散法及び，(k,L,n)しきい値秘密分散法を用いて，秘密情報から生成される分散情報（以下，Share と略す）のそれぞれに対応している対の Share を，利用者と検証者が所持している事から開始される．

認証時には，各々対の Share から秘密情報を復元し，当初設定されていた秘密情報と検証することで認証するスキームである．このスキームは，(k,n)しきい値分散法及び，(k,L,n)しきい値分散法を利用することができ，秘密情報が特定の属性情報と関連付けられた場合は，謎として利用でき，秘密情報が鍵である場合は，鍵共有として利用できる．鍵共有として利用する場合は，任意のタイミングで秘密情報を更新し，Share を配布する事で，共有鍵を更新する事ができる．つまり，予告型のワンタイム Share 認証方式である．

本稿では，該当スキームの秘密情報を権利情報として，権利認証を行う構成である．

権利認証を構成するにあたり，電子権利を以下のように定義する．[5,6,7,8]

発行者：I(Issuer)，所持者：H(Holder)，約束：P(Promise)とし，権利内容は，約束情報に内在され，電子権利 R;{I,H,P}とする．

秘密分散における秘密情報 S が電子権利 R として扱う構成であり，秘密情報 S データの選択基準は，認証スキーム[2]と同様に，一様ランダムで且つ 128bit 以上で規定され，生成される Share も一様ランダムで且つ 128bit 以上である．

以上に示したスキーム及び，データ構成によって，権利認証を行った場合の利点を書きに整理する．

1 秘匿された状態で携帯できる点．
権利情報は，携帯していても秘匿されているため，オフラインでも権利認証や権利の確認を行う事が可能である．また，権利情報自体は，分散化され，秘匿された状態で携帯される為，安全で且つモビリティ性が高い．

2 Share から権利情報を復元するだけで検証できる点．
権利情報は，Share 化されて，検証時には対になる Share を送受信し，Share から権利情報を復元するだけで検証ができる為，CHAP などの共通鍵暗号方式を利用した権利認証に比べ，効率的である．

3 検証行為は，I でも H でも可能である点．
権利情報は，権利発行者(I;Issuer)と権利所有者(H;Holder)で対の Share で保持されている為，権利発行者側で権利を検証する事も，権利所有者側で権利を確認する事も可能である．つまり，権利所有者が権利発行者から Share を入手し，権利所持者側で権利情報を確認する事ができる．

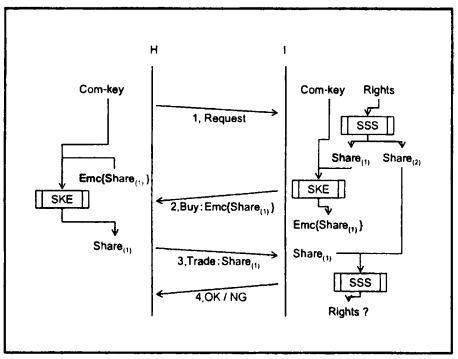

図 1：権利行使(Trade)プロトコル

3. 回数制限の実装

3.1 Trade プロトコル

図 1 は，電子権利の権利購入から，権利行使に至る一連の処理(Trade)を実現する為のプロトコルである．

発行者：I 及び，所持者：H は，上記に説明した簡易認証スキームを利用している事を前提としている為，秘密分散：SSS(Secret Sharing Scheme)，共通鍵暗号：SKE(Secret Key Encryption)モジュールが実装され，認証及び共通鍵暗号に用いる鍵(Com Key)が共有されている．

所持者：H の Request によって，発行者：I との権利発行に合意が得られた場合，

約束：P を Rights とし秘密分散する事から始まる．

この構成例では，簡単のために，(k,n)しきい値秘密分散法を用い，(2,2)で権利：Rights が分散化される事とする．

生成された $Share_{(1)}$，$Share_{(2)}$は，共有されている鍵(Com Key)を用い，共通鍵暗号(SKE)によって暗号化され，$Enc\{Share_{(1)}\}$を生成する．

権利授受として，所持者：H に配送される．

この行為は他の通信や手段を用いて有価物との交換を意味し，購入行為(Buy)である．

所持者:H は，暗号化された権利である $Enc\{Share_{(1)}\}$ を発行者：I と同じ共有鍵及び，共通鍵暗号を用いて，復号し，権利情報である $Share_{(1)}$を生成する．

実際に，権利行使(Trade)したい場合は，生成した $Share_{(1)}$を発行者：I に送信し，Share(1)と発行者：I が自ら所持している $Share_{(2)}$を用いて，秘密分散法で復元処理を行い，正当な秘密情報 S=Rights が検証された場合に権利が行使される．

この実装例は，権利行使回数が，1 回であるが，購入(Buy)の際に複数の Share を配布し，権利行使毎に 1 個づつの Share を送信するだけで実装が可能である．

実装例で示した通り，権利情報は，分散化され秘匿されている為，所持者：H は，安全な状態で権利を携帯する事が可能である．

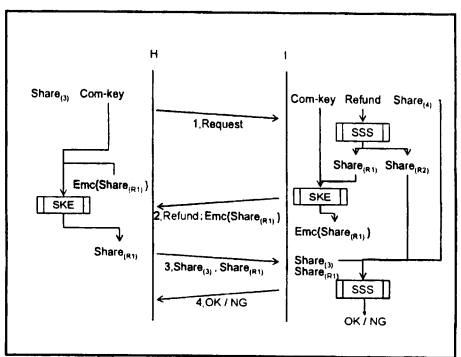

図 2：払戻し(Refund)プロトコル

3.2 Refund プロトコル

図 2 は，払戻し(Refund)処理を行うプロトコルである．前提条件は，Trade プロトコルと同じであるが，既に Trade プロトコルを用いて，権利情報である $Share_{(3)}$, $Share_{(4)}$を発行者：I 及び，所持者：H が所持している．また，Trade プロトコルと同様に，(k,n)しきい値秘密分散法を用い，(2,2)で権利：Rights が分散化される事とする．この Refund プロトコルで新たに追加される情報は，払戻し権利である Refund である．Refund データの選択基準は，認証スキーム[]と同様に，一様ランダムで且つ 128bit 以上で規定され，生成される Share も一様ランダムで且つ 128bit 以上である．所持者：H が払戻し処理を行う Request を発行者：I に送った段階で，発行者：I は Trade プロトコルと同様に，Refund データを分散化処理し，$Share_{(R1)}$と$Share_{(R2)}$を生成する．以降，Trade プロトコルと同様の手順で処理を行うが，権利を払戻しする際には，権利情報である $Share_{(3)}$と，払戻し権利である $Share_{(R1)}$を同時に送信する事で完結する．つまり，Trade プロトコルとほぼ同様のモジュール及び，プロトコル構成で払戻し処理が実装できる．

4. 権利譲渡

先に説明した通り，まとめ買い処理において，権利の譲渡が求められている．権利譲渡は，権利発行者及び，権利所持者が権利譲渡を行う事を認可しているコンセンサスが必要である．また，権利譲渡では，権利が移動される際に，権利の移動先と移動元が，各々が権利移動の合意を確証させる事が必要であり，それら合意した事が権利発行者が検証できる事が必須である．
この確証データと確証データの検証を含めたプロトコルを下記に定義する．

権利の譲渡処理は，一貫した処理で行うプロトコルであり，権利情報の移動が完結される迄にネットワーク上の問題及び，その他の障害で不正終了が発生した場合でも，権利情報が消滅しない事が前提となる．その為，権利譲渡処理を見越した Share 配布方法は，Trade プロトコルの(k,n)しきい値秘密分散法を用いた場合の(2,2)で権利：Rights が分散化されるのではなく，(2,3)というように Sahre を生成し，事前に安全な状態で復旧用の Share が保管されている事が前提となる．

図3は，権利譲渡(Transfer)を行うプロトコルであり，図 4 は，権利所持者：H が譲渡に対して合意した事(Abort)を示すプロトコルである．図 5 は，権利移動先である新たな権利所持者：H'が権利を譲渡された事(Keep)を示すプロトコルである．

権利情報を譲渡した事を確証するデータは，Abort-Commit であり，権利情報を譲渡された事を確証するデータは，Keep-Commit である．
図の中では，Abort-Commit を生成する事を「Generation;Abort-Commit」と明記し，所持者：H が，Abort-Commit を生成した後には，権利情報である Share は削除される．

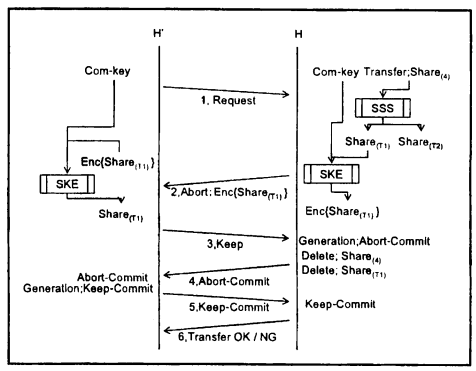

図 3：権利譲渡(Transfer)プロトコル

4.1 Transfer プロトコル

Transfer プロトコルは，Refund プロトコルと同様の前提条件で開始されるが，所持者：H は権利情報 $Share_{(4)}$を所持し，発行者：I は，$Share_{(4)}$に対応する $Share_{(3)}$を所持している．また，所持者：H と新たな所持者：H'で SKE，Com Key が共有されている事とする．所持者：H が，新たな所持者：H'に権利情報を譲渡したい場合，所持者：H は自らが所持する権利情報 $Share_{(4)}$を秘密分散処理する．生成された $Share_{(T1)}$を新たな所持者：H'に暗号化後に送信し，$Share_{(T2)}$を発行者：I に暗号化後に送信する．新たな所持者：H'は，譲渡権利情報である $Share_{(T1)}$を受信した後，権利情報を譲渡された事(Keep)が終了した事を所持者：H に返す．所持者：H は，権利情報を譲渡された事(Keep)が終了した事を確認すると，譲渡した確証データである Abort-Commit を生成し，$Share_{(4)}$及び $Share_{(T1)}$を削除する．所持者：H が生成した Abort-Commit は，新たな所持者：H'と発行者：I に送信され，新たな所持者：H'は，Abort-Commit を受けると，Keep-Commit を生成し，所持者：H と発行者：I に送信する．

以降，所持者：H と発行者：I 間では Abort プロトコ

ルが開始され，新たな所持者：H'と発行者：I 間では Keep プロトコルが開始され，全てが完結されると，権利情報の譲渡処理が完了する．

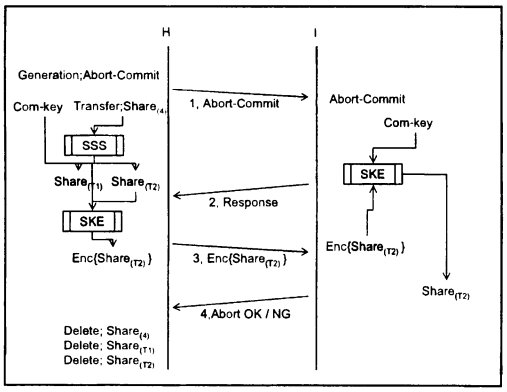

図 4：権利中止(Abort)プロトコル

4.2 Abort プロトコル

Abort プロトコルは，所持者：H が，権利情報の譲渡確証データである Abort-Commit を発行者：I に送る事から開始される．所持者：H は，譲渡権利情報の対である $Share_{(T2)}$を暗号化し，発行者：I に送り，発行者：I の受信完了後に，権利情報の $Share_{(4)}$, $Share_{(T1)}$, $Share_{(T2)}$を削除する．

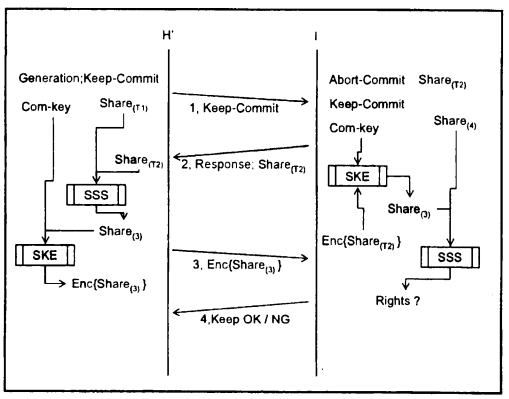

図 5：権利引継(Keep)プロトコル

4.3 Keep プロトコル

Keep プロトコルは，新たな所持者：H'が，権利情報の譲渡された確証データである Keep-Commit を発行者：I に送る事から開始される．発行者：I は，所持者：H から受信した $Share_{(T2)}$を，新たな所持者：H'に送信し，発行者：I は，$Share_{(T1)}$と，$Share_{(T2)}$から $Share_{(3)}$を生成する．所持者：H から譲渡された権利情報が正しい情報である事を確認するため，$Share_{(3)}$を暗号化し，発行者：I に送信し，確認する．

以上の構成で，権利譲渡した確認を行う Abort プロトコルと，権利譲渡された確認を行う Keep プロトコル及び，譲渡を完結する Transfer プロトコルによって，権利移動の移動先と移動元が確証を取りながら移動する事が出来る．この権利譲渡の一連のプロトコル（以下，AKT プロトコルと略す）で生成している確証情報 Abort-Commit, Keep-Commit を複数に配布する事で，信頼できる第三者(Trusted Third Party)を含めたモデルにも拡張できる．

5. まとめ

本稿では，(k,n)しきい値分散法及び，(k,L,n)しきい値分散法によって権利情報を分散化し，分散化した情報の真正性を検証する事で，権利認証を行う権利認証スキーム[2]の構成例を示し，実装評価を行った．権利発行者：I を PentiumIII-500MHz/ 256MB/Win2k, 権利所持者：H を Intel-PXA250(400MHz)/64MB/ PocketPC2002 と想定し，赤外線通信(IrDA)における権利認証プロトコルである Trade プロトコルを実装し，10 回の平均処理時間は，0.197s である．尚，計測方法は，赤外線通信のコネクト部分が終了した後，権利認証などに使用するデータを収集する前を開始点として，クライアント側がサーバから認証完了のメッセージを受け取った直後までを計測した結果である．

今後の課題は，各確証データである Abort-Commit, Keep-Commit の実装方法の検討及び，AKT プロトコルの効率化，それらを含めた計算委託の実装方法について検討する必要がある．

参 考 文 献

[1]モバイル EC-WG,"モバイル電子チケットのビジネス要件・機能要件", tech.rep.,電子商取引推進協議会, 2002.

[2]A.Shamir, "How to share a secret," comm.ACM,vol.22,no.11,pp.612-613,1979.

[3]山本 博資, "(k,L,n)しきい値秘密分散システム,"電子通信学会論文誌,vol.J68-A,no.9,pp.945-952,1985.

[4]小川　博久．山本　博資，"閾値秘密分散法を用いた簡易認証スキーム"，2003-MBL-25，Jul，2003.
[5]K.Fujimura and M.Terada. Trading among untrusted partners via voucher trading system. In proc. of the 1st Conf. on e-Commerce, e-Business, e-Government. IFIP,Oct.2002.
[6]寺田　雅之，花舘　蔵之，井口　誠，"公平な電子権利価値流通のための楽観的な交換プロトコル"，CSS2003，2003.
[7]諸井　太郎，亀山　渉，"コンテンツ流通おける認証機関を介さない権利譲渡方式の実現手法"，FIT2003，N-017，Sep，2003.
[8]福島　晋，松浦　幹太，今井　秀樹，"電子権利流通方式に関する特性分析"，SCIS2002，Jan，2002