マルチメディア通信と分散処理　52－19
（1991．9．25）

文書会議システムにおける文書認証･署名方式の検討

田中 俊昭　　中尾 康二　　鈴木 健二

国際電信電話株式会社 研究所

あらまし　現在のオフィス環境においては、ファクシミリや電話などに対してセキュリティ機能を付加したシステムが開発されているが、今後はさらにオフィスで扱われる文書の電子化に伴い、文書交換、文書検索や文書会議などの様々な文書通信におけるセキュリティ機能が重要になると考えられる。筆者らはこれまで文書通信の一形態として各端末で同一の文書を保有し、実時間で相互編集を行なう文書会議システムの検討を行ってきた。本稿では、文書会議システムに特有のセキュリティ機能の要件を明確化し、これらセキュリティ機能の要件を満足する相互相手認証、相互署名、文書アクセス管理及び文書内部アクセス制御の考察を行う。

Study on Authentication and Digital Signature for DeskTop Conference System

Toshiaki TANAKA,　Kouji NAKAO and　Kenji SUZUKI

KDD　R&D Laboratories

2-1-15 Ohara Kamifukuoka-shi Saitama JAPAN

Abstract　Considering an integrated office document communications, Security will be currently recognized as an important facility to realize document transfer, document filing/retrieval and document conference. We have been studying DeskTop Conference System (DTC) to fulfill the future office requirements on integrated document conference including bilateral remote editing. This paper firstly extracts security requirements for DTC and proposes security mechanisms of bilateral peer authentication, bilateral digital signature, document access management and inner document access control in order to realize the security requirements.

1. はじめに

現在のオフィス環境においては、ファクシミリや電話などに対してセキュリティ機能を付加したシステムが開発されているが、今後はオフィスで扱われる文書の電子化に伴い、文書交換、文書検索などの様々な文書通信におけるセキュリティ機能が重要になると考えられる。文書通信においてセキュリティが必要となる具体的な理由として

(a)正しい相手から送られた文書であることを保証する通信を行いたいこと、

(b)文書の容易な複写･改ざんを防止するため、その文書が正当な利用者により作成されたものであることを保証したいこと、

などであり、送付文書を正式文書として扱うためには、これらの要求を満足する必要がある。

一方、将来のオフィス環境においては、共同でプログラムや文書を、あるいは複数参加者による意志決定を行うといった共同作業支援が重要になると考えられており、共同作業を意識した様々なアプリケーションが考えられている。この共同作業を行うアプリケーションとして、筆者らはこれまで端末相互間で同一の文書を保有し、実時間で相互編集を行なうパソコンをベースとした文書会議システムの検討を行ってきた[1]。文書を用いた会議システムの適用例としては、①資料や論文の共同作成、②実時間で契約文書を作成し、契約締結する電子契約、③遠隔教育システムなど様々な用途が考えられ、このような適用領域においても文書通信のセキュリティ機能が必須となる。これまで、共同作業支援でのセキュリティ技術は様々な検討がなされているが[2],[3]、会議等の共同作業を意識した文書通信のセキュリティ技術は検討されていない。従って、本稿では文書会議システムに特有のセキュリティ機能を明確化し、その要求条件と実現方法について考察を行なったので以下に報告する。

2.文書会議システム

これまでの文書作成においては、ワープロ等を用いて単独に文書を作成することが一般的であったが、本論で取り上げる「文書会議システム(以後、DTC(DeskTop Conference System)と呼ぶ)」は、図1のようにワープロ等を通信回線で接続し、2者間で共同で文書を作成していく会議形式の文書通信システムで、契約書、共同会議資料、及び連名論文などの共同作成を電子的に、実時間で行うアプリケーションのためのものである。以下にDTCの特徴を示す。

(1) 2者で会議を行い、会議中は会議の相手を固定する。これにより、文書会議が終了した時点で得られる会議の出力文書は、会議参加者(2人)が共同で作成したことになる。

(2) DTCにおいて扱われる文書には、㋐白紙の文書(2者で最初から作成)、㋑事前転送された文書(会議参加者の一方が用意)、㋒以前にDTCで作成した出力文書の3種類がある。

(3) 2者で作成した文書は、全く同一である。

(4) 作成された文書は正式な出力文書としてお互いの文書ストアに登録され、その後の文書会議において再度利用する(上記(2)㋒)こともできる。

(5) 一般的な文書会議の進行形態としては、①通信環境の設定、②編集目標となる文書の選択、③編集開始、④具体的な文書編集、⑤編集終了、⑥文書登録、⑦通信終了の一連の流れがある。

(6) 会議資料の転送/編集のための通信プロトコル及び文書フォーマットは、CCITTで標準化されたDTAM(Document Transfer & Manipulation: 文書転送及び操作)[4]やODA(Open Document Architecture: 開放型文書体系)[5]をそれぞれ採用している。

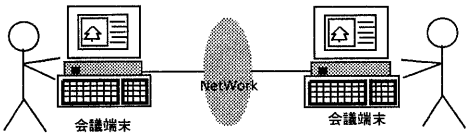


図1 DTCの構成図

3.DTCのセキュリティ機能

上述の通信システムを用いて、正式な電子契約や共同執筆を実現する場合は、以下のようなセキュリティ機能が要求される。

(1) 相互相手認証　相手が自分の意とする通信相手であることを通信環境の設定時(会議アソシエーション確立時)に互いに認証する。

(2) 相互署名　DTCを用いて会議参加者AとBが作成した文書を第3者が検査することで、両者が作成したことを証明できる必要がある。

(3) 文書アクセス管理　DTCで作成した文書は、最終的に正式文書としてお互いの文書ストアに登録されるが、登録された文書を第3者から不正にアクセス(変更等)されることを防ぐ必要がある。

(4) 文書内部アクセス制御　文書内での必要な領域にのみデータ入力する、操作者間で会議の役割(例えば、先生と生徒など)がありアクセス領域が各人で異なる、などの場合は、文書の内容に依存した操作者のアクセス制御を行う。

本稿では、これらの機能について考察を行う。

4.セキュリティ機能の実現方法

4.1相互相手認証

一般的に認証とは、図2で示すようにProver(P: 証明者)とVerifier(V : 認証者)により構成され、Pのみが保有する秘密情報を用いてPを特定し、正当なPであることをVが証明することである。

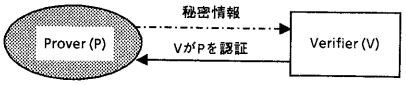


図2 認証の基本手順

DTCでは、[操作者(人間又はICカード等)]-[会議端末]-[ネットワーク]-[会議端末]-[操作者]といった通信構成をとっており、最終的には操作者間の認証が必要となる。これにより第3者が不正に侵入し、他の操作者になりすますことを防止できる。ここで、操作者間の認証を実現するには以下の構成が考えられる。

① 操作者と会議端末、会議端末と会議端末、会議端末と操作者の間でそれぞれ認証を行い、これら3リンクの認証を総合して操作者間の認証を実現する構成(図3a)。この際、会議端末は、操作者が保有する秘密情報を得て、操作者の認証を行う。

② 操作者と会議端末を一つの通信システムと考え、その通信システム間で認証を行う構成(図3b)。①と同様、操作者が保有する秘密情報を会議端末が知ることとなるが、実現手法によっては、ICカード等を用いて操作者の秘密情報をカード内で秘密処理することにより、秘密情報の会議端末への漏えいを防ぐことができる。

③ 操作者同士で直接認証を行う構成(図3c)。この場合、会議端末は認証における何の処理も行わず、操作者側が認証機能をすべて司る。この構成はICカードなど計算能力を有する機器を用いて実現できる。

これら上記の構成を比較すると、①では入力される操作者の秘密情報が会議端末内で漏えいする可能性があるのに対し、②③では会議端末が操作者の秘密情報を知らずに認証処理ができるため、②③の方がセキュリティレベルが高い。さらに、②③におけるICカードの利用においては、一般的にICカードの秘密処理の計算能力が会議端末程高くないことを考慮し、ICカードの処理負荷を最低限(秘密情報処理のみ)とし、システム間の相手認証を会議端末で行う②の負荷分散構成が望ましい。

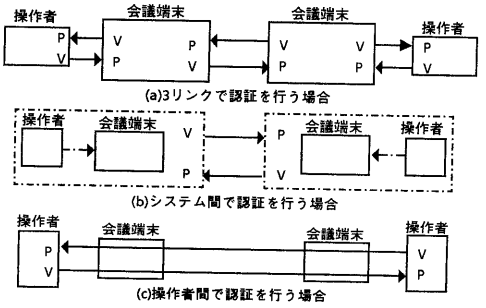


図3 相互相手認証の構成

②の構成をなす通信環境設定(会議のアソシエーション確立)時での具体的な相互相手認証のプロトコルを以下に示す。

本プロトコルは、公開系暗号を用いることにより、会議端末に操作者の秘密情報を漏えいすることなく相互相手認証を可能とする。負荷分散の形態として、操作者(ICカード)では、相手認証に用いる認証情報の作成を操作者の秘密鍵を用いて行い、会議端末では、相手認証の検証処理を相手操作者の公開鍵を用いて行う。本手順では、まず操作者が会議端末の利用を開始する際、ICカードにより会議端末が操作者の正当性をローカルに認証する手順を実行する(ローカル認証)。次に操作者が会議開始要求を行うと、会議端末がシステム間の正当性を認証する処理を行い(システム間認証)、同時に会議アソシエーションを確立する。以下にその詳細メカニズムを示す(図4)。

[前提]

会議端末 : 1, 2,

会議端末1の操作者 : A, 会議端末2の操作者 : B,

操作者AとBの識別子 : ID_a,ID_b,

操作者AとBの秘密鍵 : K_{Sa}, K_{Sb},

操作者AとBの公開鍵 : K_{Pa}, K_{Pb},

Xを鍵K_αで暗号化/復号化する : $E_{K_\alpha}(X)$,

会議端末1と2で生成されるタイムスタンプ : T_1,T_2,

ここで操作者の秘密鍵はICカード内にのみ存在する。また、公開系暗号の計算量的な安全性は保証されているものとする。

1)ローカル認証

·ICカードが挿入されると、ICカード内のID_aを会議端末1に送る。

·会議端末1は乱数R_1を生成しICカードに送る。

·ICカードは操作者Aの秘密鍵K_{Sa}を用いてR_1を暗号化し会議端末1に送る。

·会議端末1は公開鍵K_{Pa}を用いてICカードからの情報$E_{K_{Sa}}(R_1)$を復号化しR_1を得て生成したR_1と比較し、一致した場合、会議端末1は操作者Aが正しいことをローカルに認証する。

同様にして会議端末2は操作者Bが正しいことをローカルに認証する。

2)システム間認証

操作者AはBに対して会議開始要求を行う。

·$T_1\|ID_a\|E_{K_{Sa}}(T_1\|ID_a)$を会議端末2に送る[6]。ここで、$E_{K_{Sa}}(T_1\|ID_a)$の計算はAのICカードで行うとともに、会議端末1は$E_{K_{Sa}}(T_1\|ID_a)$を復号化して再度ローカル認証を行う。

·会議端末2は、Aの公開鍵K_{Pa}を用いて$E_{K_{Sa}}(T_1\|ID_a)$を復号化し$T_1\|ID_a$を得て、現在時刻とT_1を比較し、その差が許容時間内であり、かつ平文で送ら

れた$T_1\|ID_a$と比較し一致すれば会議端末2が会議端末1を認証する。

・$T_2\|ID_b\|E_{K_{Sb}}(T_2\|ID_b)$を会議端末1に送る。ここで、$E_{K_{Sb}}(T_2\|ID_b)$の計算はBのICカードが行うとともに、会議端末2は$E_{K_{Sb}}(T_2\|ID_b)$を復号化して再度ローカル認証を行う。

・会議端末1は、Bの公開鍵K_{Pb}を用いて$E_{K_{Sb}}(T_2\|ID_b)$を復号化し$T_2\|ID_b$を得て、現在時刻とT_2を比較し、その差が許容時間内であり、かつ平文で送られた$T_2\|ID_b$と比較して一致しかつT_1とT_2の差が許容時間内であれば会議端末1が会議端末2を認証し、相互認証が成立する。

[考察]　一般的に公開系暗号の計算量は多く、これをICカードで行うには時間を要し、操作性の低下や、プロトコルでの応答タイムアウト値を超える可能性がある。これを回避するため、システム間認証を行う際にICカードで計算される$T_1\|ID_a$や$T_2\|ID_b$の暗号化処理を、最初のローカル認証時に乱数と結合し($R_1\|T_1\|ID_a$など)、これを同一の操作者が利用している間、会議端末内に保持することにより、ローカル認証の回数すなわちICカードでの処理回数を最小限にする方法がある。その際T_1,T_2は、経過時間の論理性のみを検証する。これらの方法の選択は、システムの実現性を考慮して決定する必要がある。

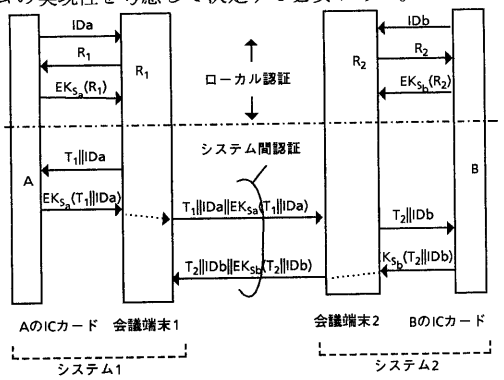


図4　アソシエーション確立時の相互相手認証プロトコル

4.2相互署名

文書は、文書全体の特徴を表す文書識別情報と実際の文書内容から構成される。文書識別情報は、文書を一意に識別する情報群であり、①文書名、②作成年月日、③作成者(複数)、④文書番号(識別子)、⑤相互署名情報などのパラメータにより構成される。一般的な文書では、契約書の締結や共同執筆を行う際に、それらの法的な正当性を確証するためには本人承認の署名と正確な作成年月日を必要とする。DTCでは、複数の作成者が実時間で文書を共同作成するため、最終的に作成された文書に対してその場で、各人の署名が必要となる。DTCで相互署名を実現するための要求条件は、

1) 相互署名した情報から確かにAが作成した文書であることを第3者が検証できること、
2) 相互署名した情報から確かにBが作成した文書であることを第3者が検証できること、
3) AとBが同一文書に対して署名を行っていることを検証できること、
4) 相互署名した情報に対して確かにAとBが作成したことを第3者が検証できること、

である。上記1)の要求条件を満足するためには、Aは保有する文書に対して縮小関数を用いてハッシュ値h_aを求め、Aのみが保有する秘密情報で暗号化することによりAの署名を作成する。2)を満足するために、同様にBの署名を作成する。また、上記3)を満足するために、システム1、2において各システムで算出したハッシュ値h_a, h_bの比較検証を行う。4)を満足するために、AとBの保有する秘密情報を両方用いて署名を作成する。以下に具体的な相互署名の作成プロトコルを示す。

4.2.1 相互署名作成プロトコル(図5)

[前提]

・A, Bが共に保有し、同一であるべき文書
　　: Ma,　Mb,
・AとBの公開鍵 : K_{Pa}, K_{Pb},
・AとBの秘密鍵 : K_{Sa}, K_{Sb},
・全システムで公開されたハッシュ関数 : h,
・Xを鍵K_αで暗号化/復号化する : $E_{K_\alpha}(X)$,
・AとBの相互署名 : S_{ab},

[step1]　各々のシステムは、共に保有する文書(M_a, M_b)からハッシュ値(h_a,h_b)を求める。

$h_a=h(M_a)$, $h_b=h(M_b)$

[step2]　得られたハッシュ値(h_a及びh_b)を自己の秘密鍵で暗号化しそれぞれ相手システムに送信する。

$E_{Ksa}(h_a), E_{Ksb}(h_b)$

[step3]　上記の相手システムからの受信情報($E_{Ksa}(h_a)$あるいは$E_{Ksb}(h_b)$)に対して、公開鍵を用いて復号化し、ハッシュ値を取り出して自己の保有するハッシュ値と比較する。一致していれば相互署名の要求条件3)の同一性が確認され、処理を続ける。不一致の場合は否定応答(NAK)を相手システムに返し処理を中断する。

[step4]　相手システムからの受信情報($E_{Ksa}(h_a)$あるいは$E_{Ksb}(h_b)$)に対して、自己の秘密鍵で暗号化し保持するとともにそれぞれ相手システムに送信する。

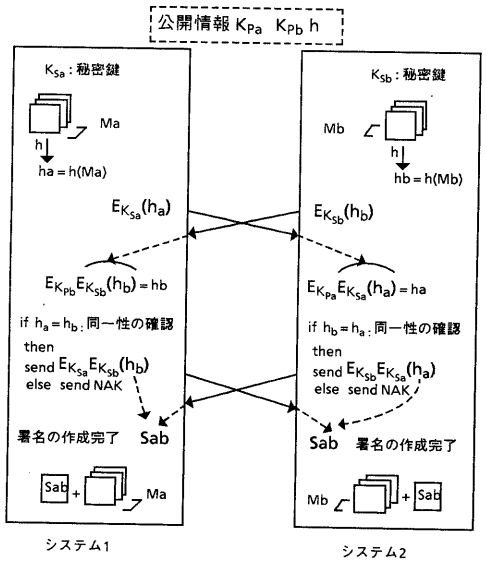
$E_{Ks_i}(E_{Ks_j}(h_j))$　　(i : 自己、j : 相手)

[step5]　step4で自己が保持している情報$E_{Ks_i}(E_{Ks_j}(h_j))$と相手から送られてきた情報$E_{Ks_j}(E_{Ks_i}(h_i))$を結合し、これを相互署名とする。

$$S_{ab}=E_{Ks_b}(E_{Ks_a}(h_a))\|E_{Ks_a}(E_{Ks_b}(h_b))$$

但し、結合情報の順序は、文書識別情報内にある作成者の順列に合わせる。例えば、上の例は文書識別情報内の作成者が"A","B"の順になっている場合である。この作成された署名情報S_{ab}を保有する文書に付加し、これを署名付きの文書M_a',M_b'とする。

$$M_a'=M_a\|S_{ab},\ M_b'=M_b\|S_{ab}$$



相互署名：$S_{ab}=E_{K_{Sb}}E_{K_{Sa}}(h_a)\|E_{K_{Sa}}E_{K_{Sb}}(h_b)$

図5　相互署名作成プロトコル

4.2.2 相互署名の検証処理(図6)

[step1]　検証者は、文書識別情報に存在する作成者のリストより、共同作成者(AとB)を確認する。

[step2]　共同作成者AとBの公開鍵K_{Pa}, K_{Pb}を用いて、相互署名S_{ab}内の$E_{Ks_b}(E_{Ks_a}(h_a))$を復号し、h_aを取り出して検証者が文書から縮小関数を用いて作成したハッシュ値h_vと比較する。一致した場合、相互署名の要求条件1)を満足する。

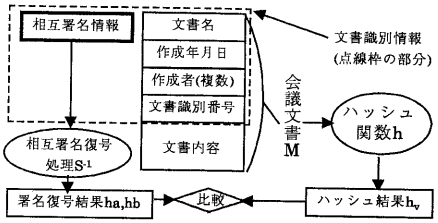
[step3]　step2と同様にして$E_{Ks_a}(E_{Ks_b}(h_b))$を復号化し、h_bを取り出してh_vと比較する。一致した場合、相互署名の要求条件2)を満足する。

[step2]及び[step3]でそれぞれの正しい署名が検証されると、結果として$h_a=h_b$が成立し、相互署名の要求条件3)が正しいことを証明できる。

[step2]及び[step3]においては、AとBの両方の公開鍵K_{Pa}, K_{Pb}を用いているので、相互署名の要求条件4)を同時に満たしている。

4.3文書アクセス管理

DTCで管理される文書群(会議文書群と呼ぶ)のアクセス権を制御することは、通常のファイル・アクセス管理と同様に重要である。会議文書群は、DTC間の



$S^{-1}=E_{K_{Pa}}E_{K_{Pb}}$ あるいは、$E_{K_{Pb}}E_{K_{Pa}}$

図6　相互署名の検証メカニズム

通信において双方の同意の基に作成された文書群を指す。図7はDTCの文書アクセス管理の例を示し、図中のディレクトリA_B及びB_Aに従属する文書群が会議文書群を表す。これらの会議文書群はその所有者であっても変更や削除することが許されない。従って、通常のOS(オペレーティング・システム)が提供するファイル・アクセス管理に加えて、会議文書群特有のアクセス権を規定する必要がある。さらに会議文書群へのアクセス権は、会議中と非会議中で異なる。

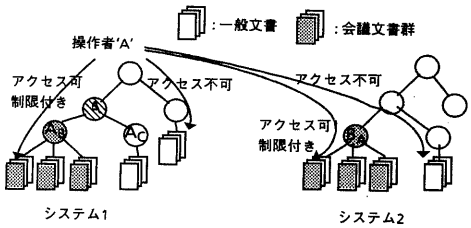


図7　DTCにおける文書アクセス管理の例

表1　操作者Aの保有する会議文書群へのアクセス権

アクセス権	読出	書込	複写	削除
操作者A	○	△	○	△
会議利用者以外の端末利用者	×	×	×	×
特権利用者(管理者)	●	●	●	●

注) ○：文書会議アプリケーションにおいてアクセスを許容されている操作
×：文書会議アプリケーションにおいてアクセスを禁止されている操作
△：会議アプリケーションでかつ、会議中は許容され、非会議中は禁止されている操作
●：特権利用者であるため、どんな操作も可能とすることができる

操作者Aの保有する会議文書群への具体的なアクセス権は、表1で示す通りである。非会議中の場合、操作者Aに「読出」「複写」を許可し、「書込(変更)」や「削除」を行うことを禁止している。このことにより、会議相手の許可なく会議文書群を操作することができない。しかし、特権利用者(管理者)はDTCに内在するどのようなファイルでも操作できるため、セキュリティの観点から問題となる。従って、このような特権利用者による会議文書群への不正

アクセスを防止するためには、非会議中/会議中にその文書を検査することにより不正な改ざんを検知する能力が必要となる。ここで非会議中/会議中で、文書改ざん検知を実現するための要求条件は、

(1) 文書の作成者(双方の操作者)による相互署名を有すること、

(2) 文書の同一性を保証する手法を保有すること、

(3) 文書を一意に識別するための識別情報を保有すること、

である。

非会議中の場合、上記(1)、(2)に対しては、4.2節の相互署名の検証処理を行うことにより実現できる((2)に対しては、相互署名の検証処理に文書同一性の検証が含まれる)。上記(3)の識別情報は文書識別情報(4.2節参照)を用いて、文書名、作成年月日、作成者(複数)、等のパラメータの論理的な正当性を検証する。

会議中の文書アクセスは、文書の選択、編集開始、具体的な文書編集、編集登録、からなる一連の処理で構成される。これらの処理の際に(1)~(3)の要求条件を満足する会議中の文書アクセス手順を示す。

a) 文書の選択の際に、文書識別情報を用いて相手システムの文書を一意に識別する。

b) 編集開始の際に、4.2で規定した相互署名の検証処理を行い、不正・改ざんが無いことを検証する。

c) 文書登録の際に、4.2で規定した相互署名作成プロトコルを用いて、新たに変更された文書に対して文書の同一性を確認し相互署名を行う(この際に、結果として相互相手認証も行われる)。

4.4文書内部アクセス制御

文書内部アクセス制御を行うためには、文書の内部を一意に識別するための文書の内部構造が必要となる。文書の内部構造とは、具体的には文書を構成する章、節、項、図など論理的な構造を指し、これらの構造を単位としたアクセス制御を行う。

会議中のアクセス権の種類としては、各操作(読出/書込/削除)に対して、それぞれ①操作者A及びBに許容される操作、②操作者Aにのみ許容される操作、③操作者Bにのみ許容される操作、④操作者A及びBに禁止される操作、に分類される。この分類に従うアクセス管理は、各論理構造に対して操作毎に操作可能な利用者リストの属性をもつことで実現できる(図8)。例えば、可能操作の一つが「読出」であってアクセスの種類が上記①の場合、「読出」可能な利用者名が"A","B"となる。しかし、非会議中では4.3節と同様に特権利用者によるOSレベルでの操作が可能となるため、不正な「読出」・「書込」・「削除」を防止するメカニズムがさらに必要となる。

不正な「書込」・「削除」に対しては、4.3節の文書アクセス管理を用いて文書全体のレベルでの不正を検出できる。不正な「読出」に対しては、秘密鍵暗号を用いて構造情報を秘匿する。ここで、不正な「読出」に対する具体的な防止法を表2に示す。表2では、秘匿を必要とする文書内部の論理構造に対して「読出」可能な操作者の秘密鍵を用いて暗号化されており、この秘密鍵を保有する操作者のみが復号化された文書を読み出せる。

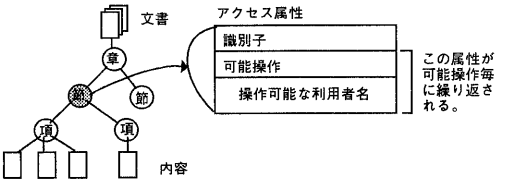


図8 論理構造が保有するアクセスのための属性

表2 不正な「読出」の防止法

アクセス権の種類	不正な「読出」防止法
①A、B許容	AとBの共有秘密鍵で暗号化
②A許容、B禁止	Aの秘密鍵で暗号化
③A禁止、B許容	Bの秘密鍵で暗号化
④A、B禁止	特権利用者の秘密鍵で暗号化
⑤すべての操作者に許容	処理せず

注)上記アクセス種類①の操作者A及びBが許容される操作は、表中ではA,Bの操作者のみに許容する①と、すべての操作者に許容する⑤に分かれている。

5.むすび

文書会議システムに特有なセキュリティ機能を整理し、それらを実現するためのICカードを用いた相互相手認証の基本構成及び相互相手認証プロトコル、相互署名のためのプロトコル及びその検証メカニズム、不正な文書改ざんを防止する文書アクセス管理方式、及び不正な操作を防止する文書内部アクセス制御方式を検討した。今後は、これらの機能を実システムで実装・評価を進めるとともに、多対地へ拡張する場合の検討を行う予定である。

謝辞 日頃熱心なご指導をいただくKDD研究所小野所長、浦野次長に感謝します。

参考文献

[1] 田中、中尾:"パソコン文書会議システムの実装・評価"情処全大(1991 3月).

[2] 高木他:"グループ指向型鍵管理に適した暗号鍵配送法"情処マルチメディアと分散処理研究会(1991 5月).

[3] Okamoto,T:"A digital Multisignature Scheme Using Bijective Public-Key Cryptosystems"ACM Trans. Vol. 6, No. 4(1988).

[4] DTAM(Document Transfer & Manipulation) CCITT T.430series.

[5] ODA(Open Document Architecture) CCITT T.410series.

[6] Entity Authentication Using a Public Key Algorithm, ISO9798-3.