H８－WG０８－１

認証局検討報告書

平成９年３月

電子商取引実証推進協議会

認証局検討ＷＧ

はじめに

インターネットのような分散したネットワーク環境において、利用者が安全かつ安心して取引や情報交換を行なえるようにするためには、取引（情報交換）に参加している当事者が真正な個人（あるいは法人）であることが証明されていなければならない。

この取引当事者が真正な個人（あるいは法人）であることを確認し、証明書（認証書）を発行する機関が認証局である。

しかしながら、認証局は誰もが運営できるわけではなく、認証局に課せられる主要な要件として、信頼性、公平性（中立性）、守秘性、相互運用性などがあり、かつこれら要件に加え組織的、経済的にも確固たる基盤を備えていなければならない。

このような種々の要件を備えた認証局を設立するための基盤整備を図り、国内外を含めた多種多様の認証局との相互運用性を確立するために、ＥＣＯＭ内に認証局検討WGが設置されている。

本WGでは企業・消費者が安心して継続的に利用できる認証局の運営を目指すとともに、最小限の認証により認証局間の相互運用が行なえる環境の構築を目的として、この目的達成のため、認証局の役割、目的等について検討を行い、認証局システムの安全対策、システムおよび財務監査、プライバシー保護等に関するガイドラインを作成し、且つ今後多種多様な認証局が設立運営されてくることに備え、認証局間の相互運用性、接続性を確保するための技術的、制度的課題について検討を行なっている。

これらの内容を検討するにあたって、認証局検討WGを３つのサブWGに分け、それぞれ以下に示す内容について検討を行なっている。

①運用制度検討サブWG

認証の信用力、継続性、安全性を保証するための広義の認証局運用ガイドラインを策定し、実証実験プロジェクトへの適用評価を行ない、実現性を高める。

②相互認証検討サブWG

多種多様の認証局間での相互認証を行なうため、公開されている相互認証技術の検討・調査を行い、相互認証基本仕様を提案するとともに、それをベースにした、日本のビジネスにおける仕組み、ルールを前提とした相互認証ガイドラインの策定を行なう。

③国際相互認証検討サブWG

国際的な相互認証を行なうため、各国における認証局の動向調査を行ない、法的、制度的課題等を

洗い出し、国際認証条件の整備を図る。

本報告書は、各サブWGの検討経過で取りまとめた「認証局運用ガイドライン」並びに「相互認証技術解説および基本仕様案」を中間成果として報告するものである。

Ⅰ部「認証局運用ガイドライン（α版）」

本ガイドラインは、公開鍵基盤における認証局の運営について、その拠り所となる指針を提示しているものであり、公開鍵基盤を構成する暗号サービス、認証書管理サービス、その他関連するサービスなど認証局が提供し得るサービスに基づき、これらに関わる認証局に課される業務要件やマネジメント要件などについて提示している。

なお、本ガイドラインはホームページにおいて平成９年１月にＥＣＯＭ会員限定で公開し、翌２月に一般に公開している。

Ⅱ部「相互認証技術解説および基本仕様案」

本書は、公開鍵基盤に基づいた認証局間の相互認証技術の調査、検討結果をまとめたものである。

既に公開されている技術の紹介と解説、加えて米国において実際に検討されている事例を解説し、加えて、これらの技術の検討結果を基に、独自に検討を行い相互認証の推奨技術として紹介している。

総合目次

Ⅰ部　認証局運用ガイドライン（α版）

Ⅱ部　相互認証技術解説及び基本仕様案

Ⅰ部　認証局運用ガイドライン

（アルファー版）

認証局検討ＷＧ
運用制度検討ＳＷＧ

Ⅰ部目次

1.はじめに

オープンなネットワークを介して行われる電子商取引においては、従来のフェース・ツー・フェースなどによる物理的な取引とは異なり、電子的に相手を確認し、取引の成約を可能にするためのデジタルな認証が重要な役割を果たす。

認証によって、ネットワーク上を流れる取引情報、更には組織内、組織間、個人間などを流れる広範囲な情報に対する盗聴、改竄、詐称などを排除しセキュリティを確保できるだけでなく、商取引に不可欠な信頼や信用を通有させることが可能になる。

本書は、そうしたデジタルな認証の発行等を行う認証局の運営において、拠り所となる指針を提示するものである。本書の構成は、1 章が認証局に関わる一般的な説明と用語の定義、2 章が認証局運営に関わる全般的マネジメント要件、3 章が認証局の各種業務に関する要件、4 章が設備･システム要件となっている。

本バージョンで対象とするのは、公開鍵に対する認証であり[1]、基本的には以下の機能やサービスの一部あるいは全てを提供する認証局についてである。なお、企業内や業界内等においてクローズされた形で運営される認証局等については、本ガイドラインを参考に個別の検討が必要である。

- 認証書の発行
- 認証書の失効
- 認証書の公開
- 認証書の保管
- ポリシーの策定/認定

本書は、今後ガイドラインの検討を進めるために取り敢えず各評価項目について基準を仮置きした叩き台であり、これを実際の基準として適用するためのものではない。評価項目の選定や、仮置きした基準についても、検討が不十分な部分が多々ある。今後は、具体的な電子商取引実験プロジェクトの経験等を踏まえ、更に検討を進めることにより、ガイドラインとしての内容の充実を図りたい。また、現在国際的な場で検討が進められている各種関連ガイドライン、例えば ISO/IEC における TTP(Trusted Third Party)のガイドラインなど、とも整合性を高めていきたい。

---

[1] 公開鍵以外にも指紋や虹彩等の生体情報や手書きサインなどに対する認証がある。しかし、本書はそれらについての配慮したものではないため、公開鍵以外の認証局運営については、本ガイドラインを部分的に適用することは可能であるとしても、全面的に適用できるようにはなっていない。

本書は、平成 8 年 5 月から活動を開始した下記グループによって、中間的にまとめられたものです。

今後も活動を継続し、本ガイドラインを充実させていく計画です。

関係各位から忌憚のないご意見、ご要望を期待していますので、下記までお寄せ下さい。

電子商取引実証推進協議会(ECOM)

認証局検討ワーキンググループ/運用制度検討サブワーキンググループ(WG08/SWG1)

〒135-73 東京都江東区青海 2-45 タイム 24 ビル 10 階

TEL：(03)5531-0065　FAX：(03)5531-0068

E-mail：yonekura@ecom.or.jp(米倉)　kakuma@ecom.or.jp(角間)

http://www.ecom.or.jp

2.イントロダクション

本イントロダクションは、用語の定義、および認証局に関連する一般的事項について概説するものであり、認証局運用に関する要件は含まれていない。ガイドラインとしての要件については、「2. マネジメント要件」以降で述べるので、基本的には、本イントロダクションを読み飛ばしても差し支えないが、一部用語の定義等については適宜参照してもらいたい。

2.1.用語の定義

１．公開鍵方式（Public Key Algorithm）

デジタル署名を作成するための秘密鍵と署名を検証するための公開鍵からなる、安全な鍵ペアを生成し利用する方式

２．公開鍵基盤（Public Key Infrastructure）

公開鍵方式を用いた情報システムセキュリティ、コミュニケーションシステムセキュリティを確保する基盤技術およびサービス。

３．公開鍵（Public Key）

公開鍵方式における鍵ペアのうちの一つ。認証局の証明により一般に公開される鍵。

４．秘密鍵（Private Key）

公開鍵方式における公開鍵と対になる鍵。自分自身で保管し、他人には公開しない。

５．認証（Certification）

取引の対象である人、サービス、情報等の真正さを証明すること。

６．認証書（Certificate）

申請者の公開鍵が真正な本人のものであることを証明するもので、申請者の氏名、申請者の公開鍵、申請者の ID、有効期間等の情報に認証局の署名を施したもの。

７．本人認証（Authentication）

事前に登録した本人であることを確認する行為。確認の方法には指紋、虹彩などのバイオメトリクスによるものやサインによるものがある。

８．認証局（Certification Authority＝CA）

申請者の公開鍵を証明し、それに基づき認証書を発行することのほか、認証書の送付、申請者の

公開鍵の登録・管理、認証局自身の鍵の生成・管理、認証書の失効登録・管理、認証局間の相互認証・接続を行なう機関。

9．登録局（Registration Authority＝RA）

エンドエンティティに関する情報（個人情報やセキュリティ情報等）の登録を行ない、認証局に対する認証書発行依頼、失効依頼を行なう。認証書の発行は行なわない。

１０．電子公証（Notary）

ネットワーク上における電子的交流の安全・信頼性を確保すること。

（電子的交流：電子メール、契約、取引の内容、電子申請、届出等）

１１．相互認証（Cross Certification）

認証局同士が相互に相手の公開鍵に対する認証書を発行しあうこと。

１２．認証書発行（Certificate Issuance）

認証書を生成し、認証書に登録された申請者に対し、その内容を通知する認証局の行為。

１３．認証実施規定（Certification Practice Statement＝CPS）

認証局の信頼性、安全性を一般に説明するために、認証局の運用、認証書発行ポリシー、鍵の生成・管理などのキーマネジメント、責任補償に関して、一連の規定を盛り込んだ文書で、ディスクロージャーの対象となる。

１４．認証書の失効（Revocation）

認証書の有効期間内に、秘密鍵の漏洩等、秘密鍵の信用性が失われたり、その恐れがある場合、あるいは申請者の氏名変更等、認証書申請時と属性が変更になった場合に認証書そのものを無効にすること。

１５．認証書の一時失効

認証書の有効期間中に一時的に認証書を失効させること。

１６．認証失効リスト（Certificate Revocation List＝CRL）

失効した認証書が登録されたリスト。

１７．デジタル署名（Digital Signature）

メッセージの送信者が公開鍵方式により、ハッシュ化したダイジェストを秘密鍵で暗号化したものをデジタル署名という。受信者が送信者の公開鍵で復号化できることにより真正な送信者であることが確認できる。

１８．デジタル封書（Digital Envelope）

公開鍵方式において電文の暗号化に用いた対称鍵を送信の際に、受信者の公開鍵で暗号化したものをデジタル封書という。

１９．エンティティ（Entity）

認証の対象となる実在する個人および法人。認証局も対象になる。

2.2.公開鍵基盤

公開鍵基盤(PKI：Public Key Infrastructure)は、電子商取引をはじめとして、情報処理システムのセキュリティやコミュニケーションシステムの信頼性を確保する上で必要となる様々なサービスのインフラストラクチャーとなるものである。

本書でいう認証書は、少なくとも利用者の名前と公開鍵(ビット列)を情報として含むデジタル文書で、認証局のデジタル署名を付したものを言う。従って認証をより正確に表現するならば公開鍵認証ということになる。

以下では PKI を構成するサービスについて概観する。

2.2.1.暗号サービス

PKI で利用される基本的な機能・技術として以下のものがある。

(1)鍵ペアの生成・保管：公開鍵/秘密鍵のペアを生成するとともに、秘密鍵はパスワード等で保護されたファイルや IC カード等のハードウェア/ソフトウェアのモジュールに保存して他人に知られないように保管する機能。なお、鍵の用途として主に以下のものが挙げられる。

・デジタル署名

・通信データ秘匿用共通鍵の暗号化

・否認防止など。

(2)デジタル署名の生成：メッセージダイジェストを生成し、デジタル署名する機能。

(3)デジタル署名の検証：メッセージとそれに対するデジタル署名が署名者のものであるかどうかを検証する機能。

(4)通信データ秘匿用共通鍵の生成･配布：通信文を暗号化するための共通鍵を生成し、それを相手に配布するための機能。

2.2.2.認証書管理サービス

PKI の中で核となる認証書の管理に関する以下のサービスであり、主として認証局によって提供されるものである。

(1)認証書の発行(Certificate Issuance)：

個人や法人等のエンドエンティティに対して認証書を発行する。発行する認証書には、階層構造の下位に位置する下位認証局(subordinate CA)、相互認証における相手の認証局等に対するものも含まれる。

(2)認証書の失効(Certificate Revocation)：

被認証者は、認証書に含まれる公開鍵と対になった秘密鍵をなくしたり、盗まれたり、あるいは解読されたりした場合、またはその可能性がある場合には、その認証書を無効にする必要がある。認証局は被認証者の確認を取った上で、認証書失効リスト(Certificate Revocation List: CRL)等によって、失効情報を利用者等の関係者に知らせる。

失効の一種に一時失効(Certificate Suspension)がある。これは、ある期間だけ失効させるものであり、一時失効の期間が過ぎれば、通常自動的に失効は解除される。一時失効は、漏洩等の可能性がある場合や、不在の場合等に利用される。

(3)認証書の公開(Certificate Publish)：

発行済みの認証書を他の人が入手できるようにするため、ディレクトリ(X.500 仕様のものやそれ以外のディレクトリなど)に登録する 。ディレクトリは認証局が管理する場合もあるし、認証局以

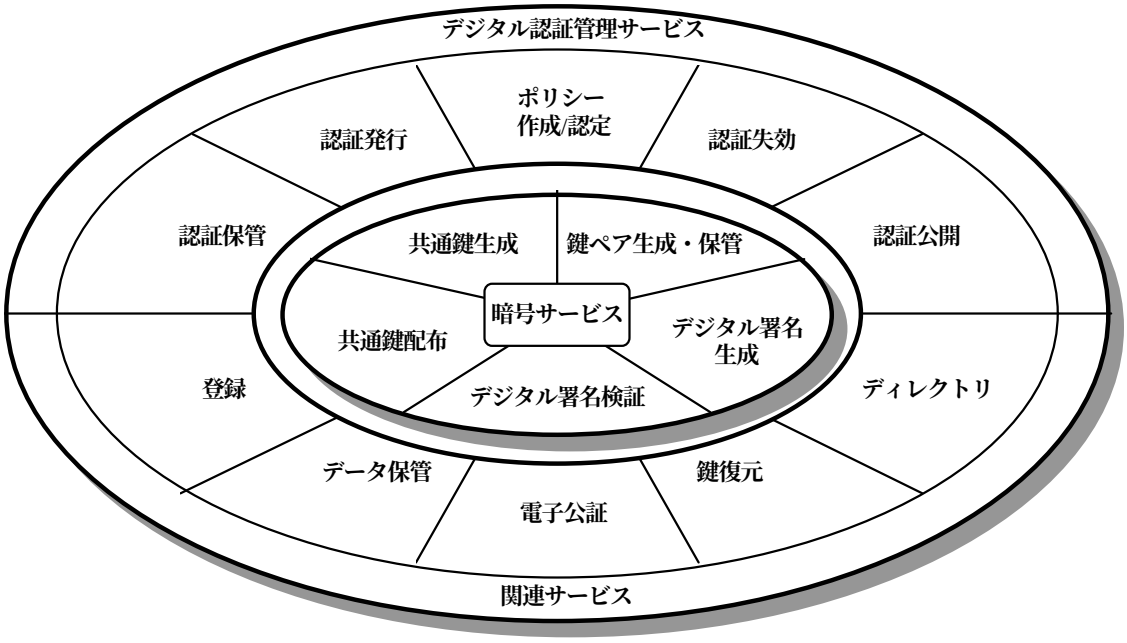


図 2.-1公開鍵基盤の構成

外の第三者が管理する場合もある。

なお、そのようなディレクトリとしては種々のアクセス制限の機能が用意され、プライバシーを守りたい場合には不特定多数に公開しないようになっているものもある。

(4)認証書の保管(Certificate Archiving)：

発行済みの認証書や CRL 等を長期にわたって保管する。これは、デジタル署名した文書自体が認証書の有効期間を超えて存在するため、それに対応させて有効期間が過ぎた認証書を、長期間保管しておく必要があるからである。

(5)ポリシーの作成/認定(Policy Creation/Approval)：

認証業務の実施に際して必要となる各種のポリシー(Policy)を定める。ポリシーには、認証局の運用に関わる要員、設備、各種手続き等を明確化した運用ポリシー、及び利用者や他の認証局等に対して認証を発行する際の審査基準等を定めた発行ポリシー等がある。

2.2.3.関連サービス

前記の認証書管理サービスに加えて、以下のような各種の関連サービスも PKI の構成要素として考えられる。これらのサービスは、認証局が付加サービスとして提供することもあるし、別の機関が提供することもある。

(1)登録((Registration)：

個人情報等を登録・管理し、認証書の発行や失効に必要な本人確認を認証局に代わって行うサービス。実際の認証書発行等は認証局が行う。

(2)データ保管(Data Archiving)：

デジタル文書等のデータを長期間にわたって保管･管理するサービス。書き換え不可能な媒体等に保管することで改竄等を防ぐとともに、媒体の陳腐化によるアクセス不能等が起こらないように適宜バックアップや保管媒体の更新等が行われる。

(3)電子公証(Notary)：

デジタル文書の公証を行うサービス。

(4)鍵復元(Key Recovery)[2]：

鍵を無くしたり、あるいは鍵をアクセスするためのパスワードを忘れたりした場合に備えて、あ

[2] ISO/IEC における TTP(Trusted Third Party)のガイドライン等、国際的議論を踏まえた検討を別途行う必要がある。

らかじめ鍵の複製を預かっておき、利用者等の要請に応じて鍵の復元を行うサービス。鍵は公開鍵方式の秘密鍵の場合もあるし、共通鍵方式の鍵の場合もある。

(5)ディレクトリ(Directory)：

個人等の属性情報を総合的に管理・提供するサービス。属性情報には、認証書ばかりでなく、電話番号、E メールアドレス等の情報が含まれる。

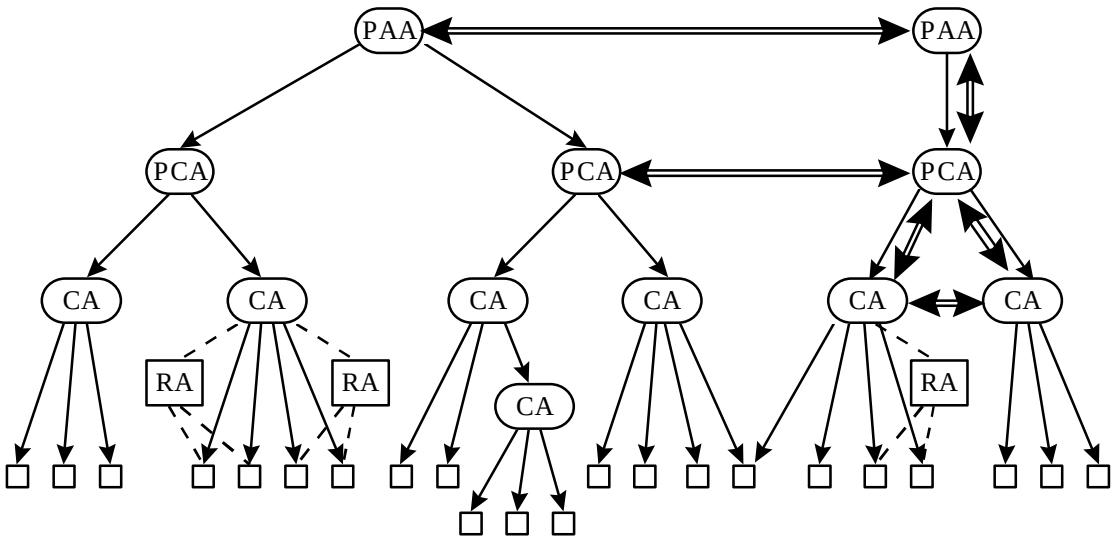
(6)その他：

鍵の保管を IC カード等のハードウェアトークンで行うような場合、鍵の生成、IC カードへの書き込みを行うサービス等のいろいろなサービスが想定される。

2.3.認証アーキテクチャー

2.3.1.構造とエンティティ

PKI における認証は、一般的に、階層構造を基本にしており、以下のようなエンティティから構成される。



PAA：Policy Approval Authority(ポリシー承認局)
PCA：Policy Certification Authority(ポリシー認証局)
CA ：Certification Authority(認証局)
RA ：Registration Authority(登録局)
□ ：End Entity(エンドエンティティ)
⟷：相互認証

図 2.-2認証のアーキテクチャー

(1)ポリシー承認局(PAA：Policy Approval Authority)

- PAA は階層構造の配下にある PCA のポリシーについて基準を定め、配下の PCA のポリシーの承認を行い、それらに認証書を発行する。PAA は PCA に対してのみ認証書の発行を行う。
- PAA は自分の公開鍵に対して自分の秘密鍵でデジタル署名した認証書(ルート認証書)を発行する。有効期間は、一般に PCA や CA のものより長い。
- ルート認証書及び PCA 等に対する認証書の発行は、原則的にオフラインで行われる。

(2)ポリシー認証局(PCA：Policy Certification Authority)

- PCA は階層構造の配下の CA のポリシーを定め、CA がポリシーを遵守した運営を行っているかどうかをチェックする。
- PCA は直下の CA に対して認証書を発行するが、基本的には直接エンドエンティティに認証書を発行しない。CA に対する認証書の発行は、原則的にオフラインで行う。
- PCA があることで、複数の CA に認証書発行を分散することができ、万が一 CA の鍵が被害を受けた場合の影響を軽減できる。
- PCA の認証書の有効期間を CA の認証書に比べて長くすることで、CA の認証書の更新を円滑に行うことが可能になる。

(3)認証局(CA：Certification Authority)

- CA は PCA の定めたポリシーに準拠して、下位 CA、エンドエンティティ及び RA に対して認証書を発行する。
- CA は PCA のポリシーに準拠して自所個別のポリシーを定める場合と、自所個別のポリシーは定めず PCA の定めたポリシーと同じにする場合がある。後者の場合は、一般的な意味での CA と区別するために発行局(IA：Issuance Authority)と言うこともある。

(4)登録局(RA：Registration Authority)

- RA は、離れた場所にいるエンドエンティティなどのために、CA への登録手続きを代行する。但し、認証書の発行は行わない。
- 登録に際しては、CA のポリシーに準拠した手続きに従い本人確認などを行う。

  RA は、LRA(Local RA)とか ORA(Organizational RA)とも呼ばれる。

(5)エンドエンティティ(End Entity)

- エンドエンティティは、認証書を利用する個人、法人、サーバなどを指す。

2.3.2.認証局のマネジメントドメイン

広義な意味での一般的な認証局は、電子メール用認証書や決済用認証書などの複数レベルの認証書を発行するものと考えられる。この場合、認証局は電子メール用認証書のポリシーを定める PCA や決済用認証書のポリシーを定める PCA 等の機能も有することになる。さらに、それらの PCA を束ねる PAA の機能を有することも考えられる。

従って、認証局がカバーする範囲(認証局のマネジメントドメインと呼ぶ)に対して、そのドメイン内におけるポリシーの統一性や整合性を保つことが求められる。また、他ドメインの認証局との間における相互認証に際しては、CA、エンドエンティティ等に対する命名規約やポリシー等の整合性を確保する必要があり、そのために予め、他ドメインのものともすりあわせ可能なように配慮しておくことも必要になる。

2.3.3.相互認証

相互認証は、認証局同士がお互いに認証書を発行し合うものであり、上記のような階層構造においては、PAA 同士あるいは PCA 同士の間、さらには CA 同士あるいは CA と PCA といった様々な組合せが考えられる。しかし、いずれの場合でも相互認証し合うもの同士の間では、ポリシーの整合性が必要になる。

2.4.認証書

2.4.1.認証書のフォーマット

認証書のフォーマットに関する国際標準として X.509 の V3 がある。この詳細については、「付録B. X.509」を参照。

2.4.2.認証書の用途と種類

認証はいろいろな用途に利用することができるが、その用途は大きく次の二つに分けられる。

(1)本人認証(Identification/Authentication)

個人や法人、あるいはサーバー等の名前や所在等についての事実であることを証明するものであり、後述する権限認証のベースとなるものでもある。何によって確認するかによって認証レベルの厳格性

に差が生じるが、3～4 段階程度にレベル分けされることが多い。

(2)権限認証(Authorization)

特定のアプリケーションに応じた権限があること証明するもの。認証の要件がアプリケーションごとに定められるものであり、前記の本人認証と異なり多様な種類が存在し得る。例えば、クレジットカード決済では、カードホルダー、加盟店、ペイメントゲートウェイ等の認証書の形式は同じであるが、それらの処理はアプリケーションに依存している。

権限認証はアプリケーションと密接に結びついており、例えばクレジットカード、銀行等のサービス提供のための一手段として利用されるのが普通である。一方、本人認証は、アプリケーションに依存せず、汎用的な利用が可能である。こうしたことから、通常、それぞれのサービスを提供する認証局は、性格が異なり、それに伴うポリシーも異なる。例えば、損害補償について言えば、権限認証では、あくまでもアプリケーションの範囲内で決まるのに対して、本人認証では認証書が何に使われるかによって補償額が大きく変動することが想定される。

本ガイドラインで主な対象とするのは、両者に共通している本人認証についてであり、アプリケーションに依存する権限認証の部分については例示や例外的な事項があることの指摘に止めている。

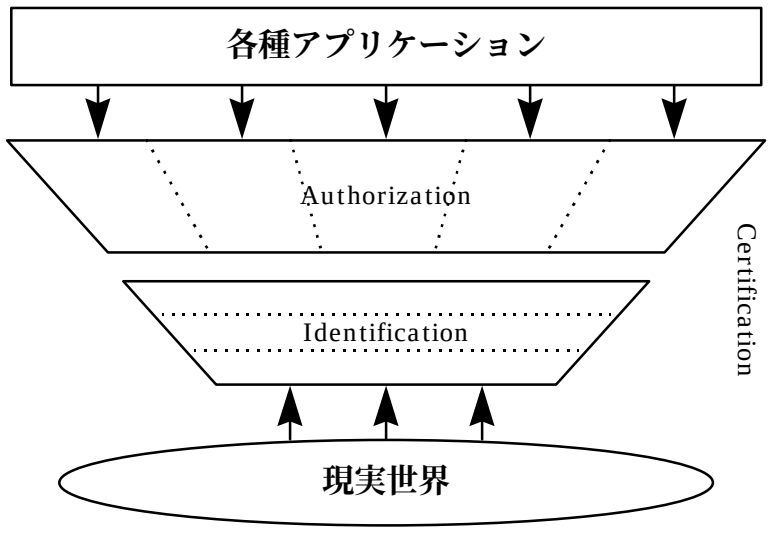


図 2.-3認証の種類

3.マネジメント要件

認証局に求められるものは信頼である。それに応えるためには、所謂人、物、金、さらには情報の面で信頼性を高めるための方策が必要である。

ところで、認証局の組織形態やサービス内容にはいろいろな種類が想定される。例えば、認証局が独立法人である場合もあるし、ある法人の一部組織である場合もあろう。また、特定の業務目的・対象にのみサービスを提供する場合もあるし、不特定多数の人々に広範なサービスを提供する場合もあるであろう。

認証局の形態によって、信頼性を高めるための方策には当然相違が出てくる。従って、以下では、共通性の高い基本的な要件について述べる。

3.1.ポリシー

認証局は、以下のようなポリシーを(方針、規程)を定め、認証を取得しようとする利用者が認証局の信頼度を評価できるように、そのポリシーを開示する必要がある。ポリシーには、大きく分けて運用ポリシー(Operational Policy)と認証書発行ポリシー(Certificate Issuance Policy)、責務ポリシー(Liability Policy)の 3 種類がある。

3.1.1.運用ポリシー

認証局にとって最も重要なものが署名用の秘密鍵であり、その管理・利用について少なくとも以下のような事項についてポリシーを定める必要がある。

- 施設・設備

  認証局のシステムが置かれている施設に対して、外部あるいは内部からの不法な侵入を防ぐための設備についてポリシーを定める必要がある。

- 認証局に秘密鍵等管理モジュール

  認証書発行等に用いるデジタル署名用秘密鍵やそれに関わるパラメタ情報等の生成、利用、保管等においては、高度のセキュリティが要求される。そうした秘密鍵等の管理に用いられる一連のモジュール(ハードウェア、ソフトウェア、ファームウェアあるいはそれらを組み合わせたもの)に対するポリシーを定める必要がある。

  秘密鍵等管理モジュールに対する不正な攻撃から内部の秘密鍵等を守るために、管理モジュール

を侵入防止設備あるいはハードウェアボックスの内に格納する等によって、物理的に保護する必要がある。なお、もし設備に侵入されたりボックスがこじ開けられたりした場合には、内部の秘密鍵等が自動的に消失するような機構を備えておくことも、より高いセキュリティを確保する上で望ましい。

秘密鍵等を管理モジュールに入力したり、あるいはそこからバックアップ等の目的で取り出したりする場合には、複数の権限のある人間がアクセスしてはじめて可能になるようにしておくことが、より高いセキュリティを確保するためには必要である。その場合、入力情報や出力情報は暗号化し、分割化することが望ましい。

- オペレータの役割分担

  認証システムや秘密鍵等管理モジュールのアクセス、さらには認証書の元データの入力、発行申請の書類の審査などに際して、どのような者にどのような権限を付与し、どのような作業を行わせるのか等についてポリシーを定める必要がある。

- 監査用ログの収集と管理

  認証書の発行申請、認証書の発行、失効の申請、認証失効リスト(CRL)の発行等のログを収集して、どれだけの頻度で誰が監査を行うのか等についてポリシーを定める必要がある。

コメント：上記のような項目をベースに、認証局の運用セキュリティについて何段階かのレベルを設定し、それぞれの要件基準を標準化することが望ましいと考えられる。具体的には、「情報システム安全対策基準」における A～C グループのようなものを、同書を参考にしながら認証局についても定めることが望まれる。WG では、最低限の要件について叩き台試案を作成したが、検討はこれからである。各方面からの意見を求めるためにも、以下に載筆しておく。

(1) レベル 1：認証システム及び秘密鍵等管理モジュールは、鍵の掛かる部屋等に設置し、権限を有する者だけがアクセスできるようになっていること。権限者は単独でも構わない。秘密鍵等をバックアップなどの目的で外部に取り出す場合には暗号化すること。

(2) レベル 2：認証局システムが設置される施設の出入り口は、こじ開けられないような物理的あるいは電子的な錠で防御されていること。

  秘密鍵等管理モジュールは、侵入防止設備あるいはハードウェアボックスの内に格納する等によって物理的に保護されていること。また、秘密鍵等管理モジュール内の秘密鍵等に対するアクセスは、少なくとも 2 人以上の権限を有する者によって可能になること。そして、権限者の識別は、PIN や指紋等による個人認証であること。

さらに、監査用ログの収集と監査を定期的に実施すること。

(3) レベル 3：上記の中信頼に加えて、秘密鍵等管理モジュールが外部的ネットワークと切り離されて運用されていること。即ち、認証書の発行等はオフラインで行うこと。

3.1.2.認証書発行ポリシー

認証発行ポリシーは、利用者に対する認証を発行する時の審査基準を規定するものであり、以下の３段階の基準を設定する。

(1) レベル 1：被認証者の名前(Subject Name)や E メールアドレス等が存在し、他に同じものがないことを確認するもの。

(2) レベル 2：個人、法人等を証明する書類(例えば住民票、パスポート等)を提出させ、その事実を電話、郵便あるいは信頼できる第三者機関(例えば、金融機関等)が提供する情報等で確認する。例えば、クレジットカードのホルダーに対して認証書を発行するような場合、クレジットカード会社はあらかじめ本人の情報を保有しており、その情報に基づいて本人認証を行なうのであればレベル 2 に該当する。

(3) レベル 3：上記との違いは、本人との直接接触を含む複数の手段で身元の確認をするところにある。

3.1.3.責務ポリシー

認証局の責任と保証に関するポリシーを定めておく必要がある。

認証局が責任を問われる場合として以下の事象が考えられ、それぞれの事象によって責任と補償の内容も変わってくる。

(1) 実施規程違反：認証局が定めた認証実施規程に違反した行為があり、それによって利用者に損害をあたえることが起こり得る。例えば内部犯罪であり、それを発見するためには内部監査などが重要となる。このような場合は、当然認証局に責任がある。

(2) 不法侵入：ハッカー等の不法侵入者により、認証局の署名鍵等が盗まれることも起こり得る。このような場合は、十分な予防措置を取っていなかったということで、やはり認証局の責任は逃れられない。

(3) 暗号解読：誰かが認証局の署名鍵を偶然あるいは故意に解読してしまうことが起こり得る。こうしたことは、認証局の知らない所で起こるため、監査などで発見するのが困難である。暗号は何

時かは破られるものであるが、破られた場合(特に、アルゴリズムそのものの解法が発見された場合)、その影響は決して小さくないとみられる。この場合でも、そのような技術を利用している認証局にも責任があるのは確かである。

いずれの場合にしろ、技術やアルゴリズム等について、そのの動向を絶えず把握しておくことは認証局の義務である。

コメント：認証局が無限の補償責任のリスクを回避するためには、米国の数州におけるデジタル署名法や ABA(American Bar Association：米国弁護士協会)のデジタル署名ガイドラインが提唱している、上限額を定めことが一つの解決策となり得る。

3.2.組織

認証局として利用者から信頼を受けるためには、以下のような組織的対応について留意する必要がある。

3.2.1.独立性/第三者性

認証局が長期的な信頼性を確保するためには、特定の企業・機関・組織の短期的な影響からできるだけ独立していることが望まれる。

また、利用者の利便性を高めるために複数の認証局が相互に接続し合う場合には、異なる認証局の利用者の信頼を得る上で、できるだけ第三者性を高めることが望ましい。

3.2.2.専門性

信頼度の高い運用を持続的に行い、技術進歩に十分対応していくために、さらにはトラブル等に迅速に対応するために、情報セキュリティ技術やシステム監査等の専門家を配置しておくことが必要である。特に、認証サービス自体、まだ揺籃期にあると言える現在、未知の問題が惹起する可能性が高く、そのような問題に迅速に対応していくためには専門的な知識・スキルを有する要員を確保しておくことが必要である。

3.3.運用面のセキュリティ

ポリシーを実務として実行していくためには、作業項目や手続き、さらにはコンテンジェンシープラ

ン等について、具体的作業が正確に行えるようにマニュアル等を整備し、それらが適正に実施されるようにマネジメントすることが必要である。

3.3.1.事務取扱要領等の規定

特に以下の観点から当該認証局の技術面、設備面でのセキュリティ対策に適合した厳密な事務取扱要領等が規定されている必要がある。

(1)セキュリティの対象となる場所へのアクセス

- 入退館、入退室管理
- 施錠、鍵管理
- 監視装置等のアクセス　等

(2)セキュリティの対象となる機器類(端末等)へのアクセス

- 端末使用権限
- カード、キー等の保管　等

(3)セキュリティの対象となる情報へのアクセス

- 情報のセキュリティレベル
- アクセス権限付与
- 媒体類の取扱い(持込み、持出しを含む)
- ドキュメント類の管理　等

(4)単一人物によるアクセスの制限

- 複数人によるアクセス
- 受付と端末アクセスの分離　等

(5)同一人への権限集中の排除

- アクセス権限と承認権限の分離
- 依頼承認手続き　等

(6)人事管理

- 人事配置(人事の固定の排除)　等
- メンタルヘルス、健康管理
- 処遇

(7)外部委託先等に対するセキュリティ上の要求

- 機密保持契約　等

3.3.2.事務取扱要領等の厳守

事務取扱要領等が厳密に守られるため以下が実施されている必要がある。

(1)内部検査、外部検査の実施

- 定期及び臨時検査の実施
- 外部委託先に対する検査　等

(2)定期的な訓練の実施

- 通常手順の実施(正しい手順の習熟、誤った手順の是正)
- 例外的(故障時、災害時等)手順の訓練の実施　等

(3)体制の確保

- 必要(規定)要員数の確保

(4)モラルアップ対策

3.3.3.監査

運用ポリシーに基づく監査を実施すること。

3.4.ディスクロージャー

認証局の健全性を維持するために認証局の持つ経営情報、技術情報、安全対策実施状況等について十分な情報開示を行う必要がある。但し、認証局のセキュリティ維持に関わる情報等については、開示するべきか否かも含めて、開示範囲をどこまでにするか十分に検討する必要がある。
また、認証局が入手した個人情報について、プライバシー等に関わるものが外部に不正流出することがないよう管理する必要がある。

3.4.1.経営情報

認証局を経営する主体は、その経営情報を開示する必要がある。例えば、法人の場合は、以下のような情報を開示する必要がある。

- 主要株主、役員
- 財務諸表

3.4.2.技術情報

認証を受けようとする利用者、あるいは既にサービスを受けている利用者の信頼度を高めるためには、必要な技術情報を適宜提供することが必要である。

3.4.3.安全対策実施状況

安全対策が十分に行われているかが判断できる情報を提供する必要がある。例えば、

- 内部監査による安全対策実施状況
- 外部監査による安全対策実施状況

3.4.4.認証実施規程(CPS)

認証の利用者が認証局の信頼性･安全性･経済性等を評価できるように、認証局のポリシーや約款、外部機関との関係などに関する詳細を記述した認証実施規程を公表する必要がある。

3.5.財務基盤

認証局を営む機関は以下の理由から充分な財務基盤を保持し運営していく必要がある。

- 認証局の責に帰される損害への賠償
- 認証局の諸機能遂行に係る継続的な投資

特に、当面継続的な投資は不可欠であり、それに必要な十分な資金を有していることが重要である。

4.業務要件

本章では、多岐にわたる認証局の業務のうち、基本的な業務である認証局の鍵管理、認証書発行、認証書の公開、認証書の保管と管理、認証書の失効と一時失効の５業務について、各々の要件を規定する。

4.1.認証局の鍵管理

4.1.1.認証局鍵ペアの生成

4.1.1.1.鍵ペアの生成

認証局の鍵ペア（署名用、通信データ秘匿共通鍵の暗号用、リポケーションリスト署名用）の生成については、

(1) あらかじめ複数の生成担当者を決めておく。

(2) 生成担当者が単独では生成できない体制にする。

(3) 監査のための生成の記録（担当者名、生成年月日、生成時刻、生成場所等）を必ず残す。

4.1.1.2.監査体制

いつ鍵を生成したか、認証局内部の人間が不正に鍵を生成していないか、鍵が持ち出されていないかなどを監査できる体制を整え、定期的な監査を実施することが必要である。なお、監査体制を整える他、鍵生成の担当者が相互に監視することも必要である。

4.1.2.秘密鍵の保管

4.1.2.1.保管環境

(1)独立した保管能力の高い専用のモジュール内に保管し、保管しているモジュールからは不正に取り出すことができない環境が必要である。

(2)モジュールへのアクセスは、限られた人間のみしかアクセスできないような設備が必要である。また、アクセスする場合には複数人とし、単独ではアクセスできないようにするとともに、必ず

その記録を残す。

(3)生成した鍵は、バックアップをとり、暗号化したうえで(1)とは別のモジュールに保管する。

4.1.2.2.有効期限が到来した鍵の扱い

有効期限が到来した秘密鍵は、セキュリティを確保するうえで、破棄することが望ましい。

4.1.3.認証局の認証書取得、公開及び保管

認証局は自らが生成した鍵ペアの公開鍵に対して認証書を取得する必要がある。

4.1.3.1.上位認証局から取得する場合

認証構造において、当該認証局に対し上位認証局が存在する場合には、その上位認証局から認証書を取得する。

4.1.3.2.自己署名による場合

当該認証局に対し上位認証局が存在しない場合には、認証局自身が公開鍵への署名を行なう。

4.1.3.3.認証書の公開

上位認証局から取得あるいは自己署名を行なった認証書は、広く一般に公開する。

4.1.3.4.公開鍵の保管

認証局の公開鍵の保管については、長期にわたりエンティティが利用可能な環境を整えておく。

4.1.4.秘密鍵の失効

4.1.4.1.失効の通知

秘密鍵の紛失、盗難、漏洩、非公式な開示など、秘密鍵の失効事態が発生した場合、認証局はこれら事案によって直接的、間接的に被害・損害を被るエンドエンティティに対し、CRL の発行によって認証要求を見合わせている旨を公開する。

なお、当該認証局が上位認証局から認証されている場合は、当該認証局が上位認証局に対し失効の通知を行なう。

失効事案の発生による信用不安を取り除くための対応策を事前に作成しておく。

また、失効となった秘密鍵を使った認証書は失効させ、原因究明に努め原因が判明した時点で公表する。

4.1.4.2.確認窓口の設置

失効事案の発生した認証局は、エンドエンティティが認証局の状況の確認を行なえる窓口を明らかにする。

4.1.4.3.漏洩のモニタリング

秘密鍵が漏洩しているか確認するため、認証書の利用状況についてサンプリングによるモニタリングなどを行なうことが望ましい。

4.1.4.4.エンドエンティティに対する認証書の再発行

認証書の再発行にあたっては、認証局側からの自動再発行はせず、エンドエンティティからの再発行要求があった場合に行なう。

4.1.5.鍵および認証書の更新

4.1.5.1.計画的更新

認証局の鍵および認証書はあらかじめ有効期間を設け、定期的に更新する必要がある。なお、鍵の有効期間の設定は認証局のポリシーによる。

4.1.5.2.失効による更新

認証局の鍵が失効した場合にも、鍵および認証書の更新を行なわなければならない。なお、鍵の更新とともに失効の通知（3.1.4.1 参照。）も速やかに行なう必要がある。

4.2.認証書の発行

4.2.1.申請受付

認証申請受付の段階で、認証局にとって重要な役割・機能として、申請者の本人確認がある。仮に、認証局の本人確認が十分でなかった場合には、他人の名前を騙って認証書を入手してしまうことが可

能となり、その人になりすまして取引ができてしまうことになるからである。

従って、認証申請方式(オンライン・オフライン)に応じて、どのような本人確認方法を準備するかが重要になってくる。

認証には、Web ブラウザ、電子メール等の認証レベル(レベル１)から高額取引、企業間取引の当事者を認証するような高度な認証レベルのものまで考えられる。例えば、クレジットカード取引の認証対象(申請者）としては、カード会員、加盟店、ペイメント・ゲートウェイの三者が想定され、カード会員はレベル２、また加盟店、ペイメント・ゲートウェイはレベル３が必要と位置づけることができる。

そして、認証のレベルに応じて、本人確認の方法も自ずと異なり、更には、認証の審査方法及び手続、審査時間も異なることから、認証の申請・受付方法にもいくつかのレベルが考えられる。具体的には、オンライン申請、オフライン申請(郵送・出頭）が考えられ以下に詳細を記す。

4.2.1.1.新規受付

新たに認証書を取得するための申請の受付で、その手順・考慮点は下記「オンライン受付」「オフライン受付」に記載の通り。

(1)オンライン受付

申請者が認証局に対してオンライン形態で認証申請を行う方式。

例えば、カード会員等個人の認証に適した申請方法である。認証局所定の申請フォームを画面上に呼び出し、入力フィールド(申請必要項目）を埋めて認証局に送信する方法がある。

オンライン申請での本人確認方法は、本人を確認するための情報を複数入力させ、その情報と認証する側（クレジットカードの場合はクレジットカード会社）が持っている情報(会員登録情報）とを照合する。それが合っているかどうかで認証書発行可否の判断をする方式である。

「本人を確認するための情報」を例示すると以下の通り。

- 生年月日
- 暗証番号(PIN)
- 自宅郵便番号
- 自宅電話番号
- 決済預金口座番号
- 母親旧姓(米国の例)　　等々、及びその組合わせが考えられる。

(2)オフライン受付

オフライン受付には郵送(書類送付)による申請受付と、申請者本人が出頭しての対面による申請受付がある。この場合には認証局所定の申請書式に必要事項を記載させるとともに、申請者が本人であることを証明する書類を提示(提出)することが必要となる。

＜本人確認のための必要書類＞

－書類送付申請方式－

・印鑑登録認証書(法人・個人)

・戸籍謄本(個人)

・商業登記簿謄本(法人)　等。

－本人出頭申請方式－

上記に加え、

・運転免許証

・パスポート

・健康保険証　等。

4.2.1.2.更新受付

認証書の有効期限到来に伴う更新申請。新規受付と同等の本人確認を行うことが必要であるが、簡便な方法として例えば、パスワードチャレンジのような方法をとることも可能である。

4.2.1.3.申請の受理および意思確認

認証申請者からの申請を認証局が受理したことを申請者に対して返答通知するとともに、併せて申請の意思確認を行うもの。

受理確認(併せて認証申請意思確認)をオンライン形態で通知するものと郵便・電話等で申請者に対して受理確認を行うものとがある。尚、受理確認は、申請時とは別の方法で行うことが望ましい。(例えばWebで受付けたものはE-mailで受理確認を行う等)

4.2.1.4.申請書類の保管

申請に関わる証拠書類として申請書類の保管が必要になる。「書類保管基準」(下記例示参照)を明確にし、厳格に運用することが望まれる。

・保存期間(例えば、有効期限到来後何年)

・保存形態

一般文書(書類)

コンピュータ　アウトプット書式

記憶媒体(ＭＴ、ディスク等)

・管理部署・責任者

・保管場所

4.2.2.審査

認証局への登録における審査は、ID チェック(本人確認)であり、商取引におけるオーソリ(与信確認)とは別基準で行われるものである。

したがって、認証局が発行する認証書によって本人であることが証明されれば目的を達する。

4.2.2.1.レベル１の審査

被認証者の名前やE-mail アドレス等が存在することを確認し、重複のチェックを行うことにより行う。

4.2.2.2.レベル２の審査

個人または法人等を証明する情報・書類等を提出させ、その事実を電話・郵便あるいは信頼できる第三者機関（例えば、金融機関等）が提供する情報等で確認する。

本人を確認するデータベースの有無により、認証局には以下の２種類のタイプが考えられる。

(1)カード会社、銀行のように既に本人確認のための情報をもつ企業が認証局になる場合

(2)本人確認に必要な情報をもっていない企業が認証局になる場合

それぞれのタイプによって審査のプロセスが違うと考えられる。

(1)カード会社、銀行のように既に本人確認のための情報をもつ企業が認証局になる場合

すでに情報を持っている為、その情報をもとに本人からの申請であることを確認することを重視すべきである。

また、特定の書式をオンライン上に作成し、申請者は必要事項をオンラインで入力する形式が考えられる。

－本人氏名

－決済口座情報

－生年月日　など

(2)本人確認に必要な情報をもっていない企業が認証局になる場合

①本人を確認するための書類を申請者から取り寄せる必要がある。

－申込書　　（例：個人の場合；本人署名・捺印、住所、生年月日など）

（例：法人の場合；法人名・住所・捺印など）

－印鑑証明

－戸籍謄本

－法人登記簿謄本　など

②本人を確認できる情報を保有している機関に問い合せることによって審査とすることも可能。

4.2.2.3.レベル３の審査

レベル２の審査に加え、本人ないし法人代表者と直接接触する等複数の手段で確認を行うことによる。

4.2.2.4.審査結果の連絡

審査の結果、認証書の発行ができない場合は、通知ないし問い合せに対する回答を行わなければならない。

4.2.3.認証書の作成

本人確認後の認証書作成は以下の手順で行う。

4.2.3.1.認証書作成の作業手順

(1)自動発行の場合

①サインをする。(この時に認証書は DB に登録される。)

②サインをした認証書の内容と、申請時の書類、本人確認書類との整合性をチェックする。

③証書書送付フェーズに移す。

(2)手動発行の場合

①審査を通った申請について、その書類の内容をデータとして入力する。

②サインをする。(この時に認証書は DB に登録される。)

③サインをした認証書の内容と、申請時の書類、本人確認書類との整合性をチェックする。

④認証書送付フェーズに移す。

4.2.3.2.セキュリティ確保のための対策

(1) セキュリティ確保のために、4.2.3.1.の手順における①と②の担当者と③と④の担当者を別にする。

(2) 内部犯行防止のため、発行に関する処理について、アクセス管理されたところに取引証跡をとる。

4.2.3.3.手続の開示について

認証書作成手続に関しては、セキュリティ上の問題があるので、開示を拒否することができる。

4.2.4.認証書の受渡し

4.2.4.1.送付の種類

(1)オンライン受渡し

認証局は必要に応じて、認証書の送付に際してセキュアな通信手段を講じる必要がある。

但し、被認証者が認証局のサーバーに受取りにいくケースもある。

(2)オフライン受渡し

宅急便、書留郵便、配達証明等の受領確認のできるものを使用する。

4.3.認証書の登録と公開

認証書の登録とは、利用者に発行した認証書を、データベースに保管し、利用者に公開することを言う。 (データベースには、認証の他に関連した情報も保管することがある)

業務上、発行記録として認証書を保管する必要があるが、これについては、『4.4.認証書の保管と管理』を参照のこと。

4.3.1.認証書の登録者

認証書の登録は、認証書を発行した認証局が登録管理を行う。

認証局は、信頼出来る機関に登録業務を行なわせてもよい。

4.3.2.認証書の公開

認証局は、登録された認証書を利用者に公開する必要があるが、公開先、公開方法、公開内容、公開期間については、認証実施規程(CPS)として明確にする必要がある。

(1)公開先

認証局は、原則として登録された認証書を公開する。

誰に公開するかは認証局 のポリシーによって定める必要がある。

利用者が公開するか否かを選択できる仕組みを入れておく必要がある。

(2)公開の方法

公開の方法としては、オンラインによる問合せへの回答が一般的であるが、公開サービスの時間帯、アクセス方法については、認証局として明確にする必要がある。

(3)公開内容

利用者の認証書をそのまま公開する必要がある。

認証書以外の個人情報を公開してはいけない。

(4)公開期間

認証書の公開は、利用者への認証書の発行後、その認証書の有効期間の間は必ず公開する必要がある。

4.4.認証書の保管と管理

認証書発行業務において、認証書を安全に保管する仕組み、および必要に応じて取り出して利用できるような管理の仕組みを持つことが必要である。

4.4.1.保管・管理業務の内容

認証局として発行した認証書全てを保管・管理業務の対象とする。また、これらの保管・管理業務対象のデータは、利用経歴がわかる形で保管する。

認証書は、申請書類との関係を明確にできる形式(シリアル番号等)で保管・管理する。

4.4.1.1.保管業務

認証局として発行した認証書全て及び、その申請書類、発行記録等の業務内容記録等を一定期間保管する。

保管にあたっては、必要に応じて保管能力の高い装置に入れたり、適切な施設内に保管するなどの措置をとる。

電子ファイルに収められたものについては、一定期間毎にバックアップをとり、世代管理および分散管理を行う。

保管方法については、文章化して明示する。また、その業務内容の記録を取り、一定期間保管する。

4.4.1.2.管理業務

保管する書類等の利用・公開にあたっては、関連主体間における了解、および認証局内での了解等、その内容に応じて複数主体・複数人の意志の統一を必要とするなどの適切な措置をとる。また、廃棄、消去などについては管理責任者の承認を必要とし、必ず管理記録をとる。

管理方法については、文章化して明示する。また、その業務内容の記録を取り、一定期間保管する。

4.4.2.保管・管理業務の形態

保管・管理業務にあたる管理者は適切に人選し、管理者それぞれのの業務範囲および責任を明確化するとともに、相互牽制体制を整備する。

平常時および非常時の責任体制・運用体制を確立する。これらは文書化して明示、周知させる。業務内容に優先順位を付け、緊急時にも円滑な運用ができるようバックアップ（データ、施設）体制を整える。

保管・管理業務を行う施設は、入退室管理を徹底するなど防犯・防災面に適切な処置を施したものとする。

機密データの取扱に関しては、常に最善の注意を払うものとする。

4.5.認証書の失効と一時失効

認証書の失効(Certificate Revocation)とは、認証書の有効期限内において、秘密鍵の漏洩や消失などの事故が起きた場合に、認証書に含まれる公開鍵を無効化することをいう。公開鍵を無効にすることによって、それに対応した秘密鍵も無効化される。

失効の一種に一時失効(Certificate Suspension)がある。これは、正規の失効と異なり、ある期間だけ失効させるもので、その期間が終わると自動的に失効が解除される。なお、一時失効の手続き等は基本的に正規の失効と同様である。

失効した認証書は、CRL(Certificate Revocation List: 認証失効リスト)や問合せ対応等の方法で利用者に公開する必要がある。

なお、クレジット等の権限認証の場合には、CRL 等を公開せず、別の手段を利用して失効の管理を行うこともある。この場合は、CRL の公開がないのに対応して、認証書自体の公開も行われないのが一般的である。

従って、以下の規定は、CRL 等による失効の公開がある場合について適用されるものである。

4.5.1.失効の主体

認証書の失効は、原則的に当該認証書を発行した認証局が行う。

なお、失効した認証書が膨大になる場合の対応としての CRL の分散管理や、高度な失効管理等のために、認証局は信頼できる機関に失効管理の一部或いは全ての機能を行わせることも可能である。

4.5.2.認証失効リストの公開

失効した認証書は CRL 等によって利用者に公開する必要があるが、公開方法や公開内容については運用ポリシー等で明確化しておくことが必要である。

4.5.2.1.公開方法

公開手段として CRL やオンライン問合せなどの方法があり、どのような手段を利用するかは発行認証書の種類や性格に応じて、適切なものを選択する必要がある。

例えば、クレジットカード業務の認証局の場合、カードホルダーや商店の認証書と決済ゲートウェイの認証書では、失効の公開方法やタイミングなどが異なることがある。通常、カードホルダーに関する CRL は公開されず、利用者であるカードホルダーの誰もがアクセスできるようにはならない。この理由の一つには、カードホルダーを失効させることは、一種のブラックリストになる可能性があるからである。また、CRL がカードホルダーに公開されない以上は、発行済みの認証がカードホルダーに公開されないのが通常である。勿論、カードホルダー本人が CRL 等に載っているかどうかを確認できるようにしておくことは、プライバシー保護のために必要である。

4.5.2.2.定期的発行

CRL の発行は、1 週間毎、1 日毎などというように定期的に行う必要がある。当該期間中に失効がない場合でも、ないことを知らせるために CRL を発行する必要がある。

どのような周期で行うかは、利用者に明確に示しておく必要がある。

4.5.2.3.失効時期

CRL には、失効した認証の番号等とともに、その認証書が失効した日時が含まれる場合がある。その場合、失効した日時が重要な意味をもつことが多い。従って、失効日時をいつにするかの基準を明確化しておく必要がある。

例えば、申請を認証局(あるいは第三者機関)が受け付けた時刻とするのか、申請者が申告した時刻とするのかなどである。

4.5.3.失効申請

失効申請するのが誰か、また失効の理由として何が考えられるかをまとめると下表になる。

表 4.-1失効申請のパターン

申請者	失効理由(例)	申請者確認の方法(例)
本人	秘密鍵の漏洩	本人署名
	秘密鍵の消失(パスワード忘れ、ファイル消去等)	発行申請と同様の手続
第三者機関	組織異動	第三者機関の署名
	不正利用	第三者機関の署名
認証局	認証局のミス	認証局が確認
	利用者の虚偽申請	認証局が確認

4.5.3.1.失効申請者

失効申請は、原則的には認証書の発行申請を行ったものが行うが、場合によっては異なる場合がある。以下の者が失効申請者となり得る。それぞれの場合について、手続き等を明確化し、利用者に開示する必要がある。

(1)本人：認証書発行に際して申請した本人、あるいはサーバ認証や法人認証の場合には、申請責任者。

(2)第三者機関：登録局(RA : Registration Authority)等の機関。失効対象の認証書が登録局を介して発行された場合などでは、失効申請は当該登録局が行う。また、発行申請を本人が行った場合でも、本人が死亡等の理由によって申請不可能の場合には、第三者機関(あるいは認証局)が手続きすることもあり得る。

(3)認証局：認証局が契約違反等があった場合に強制的に失効させる。なお、強制的に失効させる場

合は、本人に理由等を通知する必要がある。

4.5.3.2.審査

申請に対して、申請を受け付けるか否かを審査する必要がある。

例えば、悪意の第三者が他人の認証を失効させることがあり得るので、それを防ぐために審査が必要である。

失効申請者が本人で秘密鍵が盗まれたような場合、申請者本人のデジタル署名によって本人確認が可能である。しかし秘密鍵を無くした場合には、本人確認の方法としては登録の時と同等の手続きが必要になる。このように、ケースによって申請者確認の方法が異なる。審査に際しては、以下の点に留意することが必要である。

- 失効の手順・必要情報、基準をケース別に明確化して、それを開示する必要がある。
- 発行時と失効時の基準・手順は整合性をとる必要がある。
- 審査の記録を保存しておくことが必要である。また、保存期間、保存方法などについても手続きを定めておく必要がある。

4.5.4.CRL の生成・署名

失効リストの生成及び認証局による署名は、認証発行の場合と同様に内部統制が必要である。

また、CRL に載せた認証書については、登録ディレクトリ等からの削除方法、時期などについて定めておく必要がある。

5.設備・システム要件

利用者から信頼されるような設備・システムを構築するためには、基本的に「情報システム安全対策基準」のAグループに準拠するとともに、「不正アクセス対策基準」を参考にしてシステムの構築を行う必要がある。

コメント：信頼性の高い認証局を運営するための設備･システム要件として、「情報システム安全対策基準」等の項目から必要なものを選択し、認証局用のサブセットを定義する必要があると考えており、今後その作業に着手する計画である。

6.付録 A.参考文献

(1) 電子商取引に関する検討課題について(電子商取引環境整備研究会中間報告), 通商産業省, 1996.4
"http://www.ecom.or.jp/miti/press960423.html"

(2) 暗号認証技術を利用した鍵管理システムの調査研究, 認証実用化実験協議会(ICAT), 1996.3.14
[ICAT ホームページ: "http://www.icat.or.jp/interauth-index.html"]

(3) 情報システム安全対策基準解説書、(社)情報サービス産業協会、1996．10

(4) コンピュータ不正アクセス対策基準解説書、（財）日本情報処理開発協会、1996．11

(5) 暗号政策と電子現金（電子決済、電子現金とその利用環境整備に関する調査研究会報告書）、郵政省電気通信局、1996．4

(6) 電子決済におけるセキュリテイに関する調査研究中間報告、金融情報システム NO．172　1996．6

(7) Bruce Schneier, E-MAIL SECURITY, 邦訳：力武 健次監訳「E-Mail セキュリティ」, オーム社, 1995.5.25

(8) ISO/IEC 9594-8 : 1995| ITU-T Recommendation X.509(1993E), Information Technology - Open Systems Interconnection - The Directory: Authentication Framework, 1993.11
[案内："http://www.itu.ch/itudoc/itu-t/rec/x/x500up/x509_27505.html"]

(9) ISO/IEC 9594-8 | ITU-T Rec. X.509, Amendment 1(Final Draft), 1996.6.30
"ftp://NC-17.MA02.Bull.com/pub/OSIdirectory/Certificates/Certificates30June.US.ps"

(10) ISO/IEC DIS 11770-1:1996(E) Information technology - Security techniques - Key management - Part 1: Framework, 1996

(11) ISO/IEC CD 11770-3:1996(E) Information technology - Security techniques - Key management - Part 3: Mechanisms using asymmetric techniques, 1996

(12) ISO/TC68/SC2 N-543, Certificate Management for Financial Services(NWI Proposal), 1996.1.9

(13) ISO/IEC JT1/SC27 WD 14516-1, Guidelines for the use and management of Trusted Third Party services - Part 1: General Overview, 1995.11

(14) ISO/IEC JT1/SC27 WD 14516-2, Guidelines for the use and management of Trusted Third

Party services - Part 2: Technical aspects, 1996.6.21

(15) Housley, W. Ford and D. Solo, Internet Public Key Infrastructure Part I: X.509 Certificate and CRL Profile, IETF Internet Draft, 1996.6
"ftp://ftp.ietf.org/internet-drafts/draft-ietf-pkix-ipki-part1-02.txt"

(16) Farrell, C Adams and W. Ford, Internet Public Key Infrastructure Part III: Certificate Management Protocols, IETF Internet Draft, 1996.6
"ftp://ftp.ietf.org/internet-drafts/draft-ietf-pkix-ipki3cmp-01.txt"

(17) Bob Blakley and the APKI Working Group, Architecture for Public-Key Infrastructure, IETF Internet Draft, 1996.11
"ftp://ftp.ietf.org/internet-drafts/draft-ietf-pkix-apki-00.txt"

(18) Federal Public Key Infrastructure Technical Specifications Part A: Requirements, NIST, 1996.1.31
"http://csrc.ncsl.nist.gov/pki/require5.ps"

(19) Federal Public Key Infrastructure Technical Specifications Part B: Technical Security Policy, NIST, 1996.1.24
"http://csrc.ncsl.nist.gov/pki/tspolicy.ps"

(20) Burr, Federal Public Key Infrastructure Technical Specifications Part C: Concept of Operations, NIST, 1996.2.12
"http://csrc.ncsl.nist.gov/pki/conops.ps"

(21) Federal Public Key Infrastructure Technical Specifications Part D: Interoperability Profile, NIST, 1995.9.27
"http://csrc.ncsl.nist.gov/pki/cross.ps"

(22) The 1994 Mitre PKI Study Final Report, NIST
"http://csrc.ncsl.nist.gov/pki/mitre.ps"

(23) Warwick Ford, A Public Key Infrastructure for Unclassified but Sensitive Applications, NIST, 1995.9.1
"http://csrc.ncsl.nist.gov/pki/fordrept.ps"

(24) Santosh Chokhani and Warwick Ford, The Certificate Policy and Certification Practice Statement Framework(Draft), NIST, 1996.11.3

"http://csrc.ncsl.nist.gov/pki/docs/fmk03nov.doc"

(25) Security Requirements for Cryptographic Modules[FIPS PUB 140-1], NIST, 1994.1.11
"http://www.itl.nist.gov/div897/pubs/fips140-1.htm"

(26) Digital Signature Guidelines, American Bar Association, 1996.8.1
[**案内：**"http://www.intermarket.com/ecl/"]

(27) ICE-TEL, Architecture and General Specifications of the Public Key Infrastructure, 1996.9
"http://www.darmstadt.gmd.de/ice-tel/deliverables/download/D1-Architecture.rtf"

(28) VeriSign Certification Practice Statement, VeriSign, Inc., 1996.8.7
"http://www.verisign.com/repository/CPS/"

(29) Utah Digital Signature Act(1996)
"http://www.gvnfo.state.ut.us/ccjj/digsig/dsut-act.htm"

(30) Bradford Biddle, Misplaced Priorities: The Utah Digital Signature Act and Liabilty Allocation in a Public Key Infrastructure(Draft: October 18,1996)
"http://www.softwareIndustry.org/issues/docs-org/digsig.pdf"

(31) Christopher Kuner **英訳**, German Draft Digital Signature Law (SigG), 1996
"http://ourworld.compuserve.com/homepages/ckuner/digsig.htm"

(32) Michael Froomkin, The Essential Role of Trusted Third Parties in Electronic Commerce (Draft), 1995.4.22
"http://www.law.miami.edu/~froomkin/articles/trusted.htm"

[**関連インデックス**]

(1) Public Key Infrastructure References(PKI **関連**)
"http://www.zoo.net/~marcnarc/PKI/References.htm"

(2) CommerceNet PKI Task Force(PKI **関連**)
"http://www.commerce.net/work/taskforces/pki/pki.html"

(3) Software Industry(**デジタル署名関連**)
"http://www.softwareIndustry.org/issues/1digsig.html"

(4) The Cryptography Project(**暗号関連**)
"http://www.cosc.georgetown.edu/~denning/crypto/"

7.付録Ｂ．X.509 フォーマット

7.1. 認証書フォーマット[3]

認証書(Certificate)は、下表に示す項目からなる情報に対してデジタル署名したものである[4]。なお、バージョン 1 で定められた項目は必須[5]であるが、それ以外のバージョンで定められた項目はオプションである。

表 付録-7.-1X.509 認証書フォーマット

Version	項目	説明
V-1	version	バージョン(0 は V-1、2 は V-2 を示す)
	serialNumber	認証番号
	signature.algorithmIdentifier algorithm parameters	署名方式
	issuer	認証発行局名(Distinguished Name 形式[6]) ● 国名(country) ● 地域(locality) ● 組織(organization) ● 所属(organizationalUnit) ● 名前(commonName)
	varidity notBefore notAfter	有効期間 ● 開始日時 ● 終了日時
	subject	被認証者名(Distinguished Name 形式)
	subjectPublicKeyInfo algorithm subjectPubkicKey	被認証者の公開鍵情報 ● 鍵のアルゴリズム ● 鍵(ビット列)
V-2	issuerUniqueID	認証発行局の固有 ID
	subjectUniqueID	被認証者の固有 ID

[3] 出所：ITU Rec. X.509 | ISO/IEC 9594-8　Final draft(1996.6.30)。

[4] これらの項目全体に対するデジタル署名が authrityKeyIdentifier で定義された鍵で行われる。

[5] 被認証者名(Subject)は、従来はグローバルにユニークであることが必要な必須項目であったが、V-3 からオプションになった。

[6] Distinguished Name 形式は、国名、地域、名前等の組合せて、一つのユニークな識別名を作るものであり、識別名の重複は認められない。

Version	項目	説明
V-3	authorityKeyIdentifier keyIdentifier authorityCertIssuer authorityCertSerialNumber	当該認証の署名確認に用いるべき鍵(認証)の識別子 ●鍵識別子(8 進数) ●認証発行局名(GN 形式[7]) ●認証番号
	subjectKeyIdentifier	被認証者が複数の鍵を持つ場合の識別子(鍵の更新時などに利用)
	keyUsage (0) digitalSignature (1) nonRepudiation (2) keyEncipherment (3) dataEncipherment (4) keyAgreement (5) keyCertSign (6) cRLSign	公開鍵の利用目的(ビット列) (0) デジタル署名用 (1) 否認防止用 (2) 鍵の暗号用 (3) 電文の暗号用 (4) 共通鍵の配布用 (5) 認証の署名確認用 (6) CRL の署名確認用
	privateKeyUsagePeriod	当該認証の公開鍵に対応する秘密鍵の有効期間。通常公開鍵の有効期間より短い。署名用の鍵だけが対象。
	certifficatePolicies policyIdentifier policyQualifiers	認証発行局のポリシー(以下の複数組合せ) ● ポリシーID(ISO/IEC9834-1 に準拠) ● 認証基準
	policyMappings issuerDomeinPolicy subjectDomeinPolicy	CA 認証の場合のみ。発行認証局のポリシーと被認証 CA のポリシーのどれとどれが同一かを規定。
	supportedAlgorithms algorithmIdentifier intendedUsage intendedCertificatePolicies	ディレクトリ(X.500)のアトリビュートを定義。コミュニケーションの相手がディレクトリ情報を利用する場合に、事前にそのアトリビュートを知らせるため。
	subjectAltName otherName rfc822Name dNSName x400Address directoryName ediPartyName uniformResourceIdentifier iPAddress registeredID	被認証者の別名(GN 形式)。任意のものを選択。 ● 任意の名前 ● e-mail アドレス ● ドメイン名 ● O/R アドレス(X.400 originator/recipient address) ● ディレクトリ名 ● EDI 用の名前 ● WWW 用の URL ● IP アドレス ● 登録済みオブジェクトの ID(ISO/IEC9834)
	issuerAltName	認証発行局の別名(上記と同様)
	subjectDirectoryAttributes	被認証者の任意の属性。例えば、郵送先、電話番号、顔写真(イメージデータ)等の情報。

[7] GN(General Name)形式は、subjectAltNames の項で利用されているように、一つのエンティティに対して複数の識別情報を与えるもの。Distinguished Name 形式と異なりユニーク性は要求されない。

Version	項目	説明
	basicConstraints cA pathLenConstraint	当該公開鍵が、認証局の署名用かエンドエンティティ(認証を発行できない)のものかの区別。 ● 認証局かエンドエンティティの区別 ● 下位に来る認証局のパスの長さの制限(0 の場合は、エンドエンティティの認証のみ)
	nameConstraints permittedSubtrees base minimum maximum excludedSubtree	被認証者が認証局である場合(CA 認証)にのみ使用。 上記の basicConstraint で括ったパスの範囲内にある下位 CA について、詳細に認証通用の範囲を名前で規定。 ● 通用可能な下位 CA 及びその配下の階層範囲 －下位 CA の名前(GN 形式) －通用可能な階層範囲の上限 － 〃 下限 ● 通用除外の下位 CA(指定方法は上記と同じ)
	policyConstraints policySet requireExplicitPolicy inhibitPolicyMapping	上記に加えて、ポリシーに対する制約。 ● 必要なポリシーID ● パス長(パス長が指定値を超えた認証にはポリシーの明示が必要。ポリシーマッピングでも可) ● ポリシーマッピング不可のパス長
	cRLDistributionPoints distributionPoint reasons unused keyCompromise cACompromise affiliationChanged superseded cessationOfOperation certificateHold cRLIssuer	CRL の配布場所。 ● 配布局名(GN 形式 s)。省略時は cRLIssuer ● 上記配布局が対象とする失効理由(ビットの on/off) (0)未使用 (1)エンドエンティティの鍵が危害を受けた (2)CA の鍵が危害を受けた (3)認証の情報(被認証者の名前等)が変更(危害なし) (4)当該認証が置換えられた(危害なし) (5)利用中止 (6)利用の一時中止 ● CRL の発行局名。省略時は発行 CA

7.2. CRL フォーマット

表 付録-7.-2CRL フォーマット

Version	項目	説明
V-1	signature.algorithmIdentifier	署名方式
	issuer	CRL 発行局名(Distinguished Name 形式)
	thisUpdate	当該 CRL の発行日時
	nextUpdate	次回の発行予定日時
V-2	version	バージョン番号(ない場合は V-1、1 なら V-2 を示す)
	authorityKeyIdentifier keyIdentifier authorityCertIssuer authorityCertSerialNumber	当該 CRL の署名確認に用いるべき認証の識別子 ●鍵識別子(8 進数) ●鍵の認証発行局名(GN 形式) ●認証番号
	cRLNumber	CRL の発行通し番号
	issuingDistributionPoint distributionPoint onlyContainsUserCerts onlyContainsCACerts onlySomeResons indirectCRL	当該 CRL の配布局と性質 ● 配布局名(GN 形式) ● エンドエンティティの失効専用(の場合に「真」) ● CA 認証の失効専用(の場合に「真」) ● 幾つかの失効理由による(理由フラグをセット) ● 失効理由等の情報は、CRL 発行局でなく認証発行局に迂回。後記の certificateIssuer の項を参照。
	deltaCRLIndicator	当該 CRL がデルタ CRL かどうかの識別。ベース CRL の cRLNumber を指定することで、それに対する変化分だけを取扱う(両者の発行日時はベース CRL の方が早い)。[8]
V-1	[以下の項目を 1 組として失効 認証の数だけ繰り返す]	
	certificateSerialNumber	認証番号
	revocationDate	失効申請受理日時
V-2	reasonCode unspecified keyCompromise cACompromise affiliationChanged superseded cessationOfOperation certificateHold removeFromCRL	失効理由(以下のコードを指定) (0)理由不明 (1)エンドエンティティの鍵が危害を受けた (2)CA の鍵が危害を受けた (3)認証の情報(被認証者の名前等)が変更(危害なし) (4)当該認証が置換えられた(危害なし) (5)利用中止 (6)利用の一時中止 (8)一時中止状態の解除(デルタ CRL の場合に利用。ベース CRL で上記(6)の状態のものを削除する)
	holdInstructionCode	一時利用中止に対する対処方法(オブジェクト ID を指定)

[8] 本来、CRL は失効認証全てを蓄積したもの(ベース CRL と呼ぶ)であるが、失効数の増大によるパフォーマンス悪化の防止のために、CRL の配布元の分散と、デルタ CRL が考案された。デルタ CRL はベース CRL を基に作成されるものであり、ベース CRL は不可欠である。

Version	項目	説明
	invalidityDate	秘密鍵が危害にあったと考えられる日時。認証局がCRL を発行した日時(thisDate)より一般に早い。申請ベースなので、これだけでは否認防止(nonrepudiation)に不十分。
	certificateIssuer	認証発行局名(GN 形式)。indirectCRL の場合には失効情報が CRL 発行局で管理されていないため、指定された CA に迂回する。省略された場合は、直前のエンティティと同じ CA(最初に省略された場合は、CRL 発行者)。

8.付録Ｃ．検討メンバーリスト

ＥＣＯＭ

米倉 昭利　　主査　　電子商取引実証推進協議会 主席研究員

長　博連　　副主査　　電子商取引実証推進協議会 主席研究員

角間 和博　　副主査　　電子商取引実証推進協議会 主席研究員

有識者

大山 永昭　　東京工業大学　教授

須藤 修　　東京大学　助教授

リーダー・サブリーダー

船越 亘　　リーダー　　株式会社富士通システム総研 研究開発部 主席研究員

佐藤 裕之　　サブリーダー　さくら銀行 ネットワーク業務部 調査役

磯貝 和久　　サブリーダー　ユーシーカード株式会社 ＥＣ事業部 ｱｼｽﾀﾝﾄﾏﾈｰｼﾞｬｰ

メンバー

青木 泰　　財団法人金融情報システムセンター 安全対策部 課長

荒牧 英樹　　エヌティティデータ通信株式会社 技術開発本部

井上 清司　　沖電気工業株式会社 情報システム事業本部 担当係長

岩下 直行　　日本銀行 金融研究所 副調査役

内田 勝也　　安田火災海上保険株式会社 システム管理部 課長

大谷 彰宏　　三菱電機株式会社 Ｃ／Ｓネットワークシステム部

大橋 哲也　　株式会社ジェーシービー 企画部 調査役

加藤 丈博　株式会社ミリオンカード・サービス 営業企画部

河崎 克也　　社団法人日本クレジット産業協会 会員部 主任

北野 健二　　セコム株式会社 通信技術推進室 主任

久林 靖孝　　株式会社スフィンクスセンター

木暮 素史　　株式会社ディーシーカード マルチメディア企画室 室長

佐藤 順一　　日本信販株式会社 企画本部マルチメディア推進室 チーフマネージャー

鈴木 雅人　　日本ベリサイン株式会社 エンジニアリンググループ エンジニア

勅使河原 元 株式会社野村総合研究所 サイバーコマース事業部

疋田 時久　株式会社野村総合研究所 サイバープラットフォーム事業部

福村 和悦　日本電気株式会社 ネットワーキング技術研究所インタネット技術部 部長

藤尾 真嗣　株式会社住友クレジットサービス マルチメディア推進部

藤野 欣哉　アメリカン・エキスプレス・インターナショナル 業務企画部 副部長

光永 聖　株式会社日立製作所 情報システム部 主任技師

安國 弘晃　株式会社アドバンス 情報通信事業本部 技術開発部 次長

山田 慎一郎 日本ベリサイン株式会社 取締役営業本部長ビジネス企画部長

吉田 正敏　株式会社シー・アイ・シー 電子研究プロジェクトチーム サブマネージャー

II部　相互認証技術及び基本仕様案

認証局検討ＷＧ
相互接続検討ＳＷＧ

Ⅱ部目次

1.はじめに

この技術資料は電子商取引実証推進協会（ＥＣＯＭ）の認証局検討ワーキンググループWG８の相互接続検討サブワーキンググループの参加メンバを中心にまとめた技術資料である。

ＥＣＯＭの活動内容の一つとして電子商取引（ＥＣ）を実現する共通プラットホームとなる技術開発があり、このためのワーキンググループの一つとして認証局検討WGがある。

認証局検討WGは ①運用制度検討ＳWG ②相互接続検討ＳWG ③国際相互認証ＳWGの３つのＳWGから構成されている。

相互接続検討ＳWGでは認証局相互の連携を可能とするための相互認証技術を調査､検討している。

この技術資料は上記の相互認証技術の調査、検討結果をまとめている。公開されている技術の紹介と解説を中心として、さらに独自に相互認証の推奨技術を解説している。

この資料の対象とする読者は認証局の運用管理者である｡認証局の相互認証を行う場合の技術的な解説をまとめている。共通技術編、個別技術編、参考資料編の３部から構成されている。

まず最初に共通技術編から読まれる事をお薦めしたい｡個別技術編は代表的な認証プロトコルを対象とした個別技術の解説である。

この資料では、認証書の形式定義等でＡＳＮ.１表記をそのまま原典から引用している。

ＡＳＮ.１表記法がよくわからない場合にはこの部分を読み飛ばして頂きたい。実際にサービスをすることになった場合には、技術者の支援を受けて頂きたい。

平成９年３月

電子商取引実証推進協会

認証局検討ワーキンググループ

相互接続検討サブワーキンググループ

2.共通技術編

2.1.認証技術概要

世界中のユーザを対象とした商業ベースのインターネット利用が１９９５年から米国を中心として本格的に行われる様になってきた。このような商業ベースのインターネット利用形態をエレクトロニック･コマース（ＥＣ）と呼び、一般消費者を対象としたオンライン注文･決済から企業間の取引（ＥＤＩ）を対象としたインターネットＥＤＩなどが実用化に向けて実験が開始されつつある。インターネットを商業ベースで利用することにより安価な通信料金を享受することができると期待されているが、その反面、プライバシー保護や本人認証などのセキュリティ確保を前提としたシステム構築が不可欠となっている。

このようなセキュリティを提供するための基盤技術として**公開鍵**によるデータの暗号鍵の保護やディジタル署名による本人認証などがセキュリティを確保するデファクト技術である。

この公開鍵の技術はどのような通信の組み合わせでも一つの公開鍵のペアを使用することができるという利点をもっているが､公開鍵が確かにその本人のものであることを信頼できる第三者が保証することが前提である。このような公開鍵の正当性を保証する第三者機関を**認証局**（ＣＡ）という。

公開鍵の正当性を保証するためのディジタルドキュメントを**認証書**（Certificate）という。

認証書の形式として、ＩＴＵ/ＩＳＯ/ＩＥＣで標準化されたX.５０９形式の認証書がデファクトとなっている。

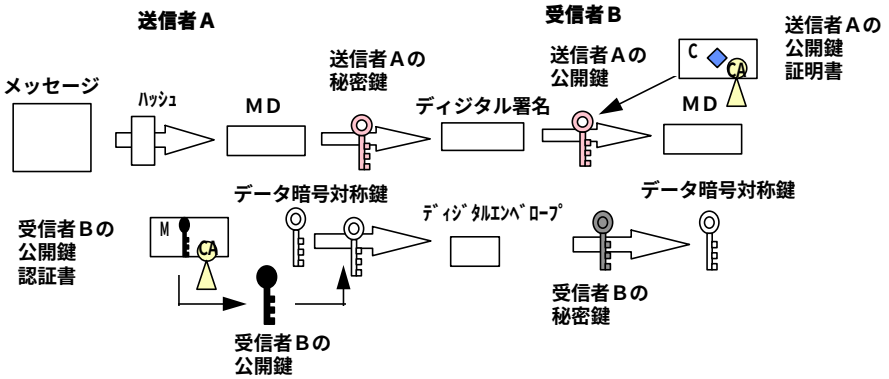


図１－１ 認証技術概要

2.1.1. 認証技術の標準化状況

① X.５０９ Version ３の仕様をＩＳO/ＩＥＣ ＪＴＣ１/ＳＣ２１/WG４及びＩＴＵ－Ｔ Ｑ１５/７で共同仕様化完了。（最新版：1995）

② ＩＥＴＦ ＰＫＩＸ WGでX.５０９認証書プロファイルと管理プロトコルの標準化中。

（最新版:1996/06）ftp://ds.internic.net/internet-drafts/draft-ietf-pkix-ipki-pat1-02.txt

（最新版：1996/12）ftp://ds.internic.net/internet-drafts/draft-ietf-pkix-ipki3cmp-01.txt

③ ＮＩＳＴで米国連邦政府のＰＫＩを標準化中。 http://csrc.nist.gov/pki/

－Minimum Interoperability Specifications for PKI Components (MISPC)

－Technical Specifications Part A-- Part D

2.1.2. 認証技術の業界動向

① クレジットカード決済仕様ＳＥＴを Visa と Mastercard が共同開発。

ドラフト仕様を公開中でこの中で認証仕様も規定。

http://www.mastercard.com/set/set.htm

http://www.visa.com/cgi-bin/vee/sf/set/intro.html?2+0

② トランスポート層のセキュリティプロトコルＳＳＬを Netscape Commnications が開発。

ＳＳＬ 3.0 でクライアント認証仕様も規定。

http://www.netscape.com/newsref/ref/netscape-security.html

2.2.相互認証の前提条件

2.2.1. システム構成と前提条件

この技術資料は、公開鍵をベースとする公開鍵技術基盤を前提としたセキュアなオンラインデータ通信環境へ適用する。このような公開鍵環境（ＰＫＩ）ではつぎのようなシステム構成を前提としている。

① 公開鍵を利用したセキュアな通信を実行するユーザとそのエンドシステム

② 公開鍵の信頼性を保証するための認証書を発行する認証局

③ 認証局の信頼性を保証するための上位認証局

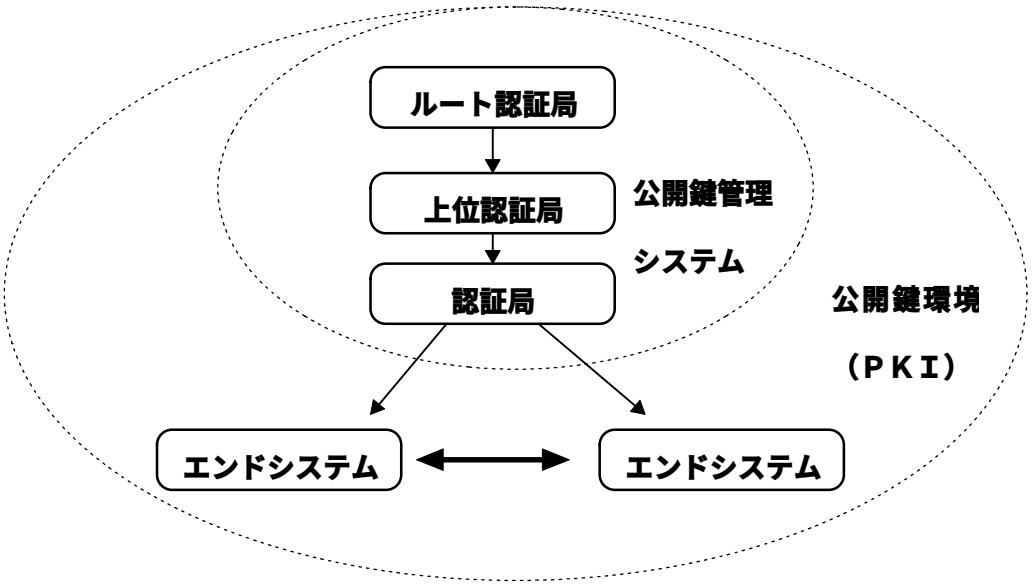


図１－２ 前提とするシステム構成

ＰＫＩで使用する認証書のフォーマットとして幾つかの標準フォーマットがあるが、この資料では、Ｘ.５０９認証書を標準フォーマットの前提とする。

2.2.2. 本資料の目的

この資料は、上記の公開鍵管理システムを運営する管理者が異なる管理ドメインの認証局間の相互認証を検討するための技術的な情報を提供する事を目的としている。

相互認証においては認証局相互のポリシーの整合性を図る事を前提にすることが重要である。

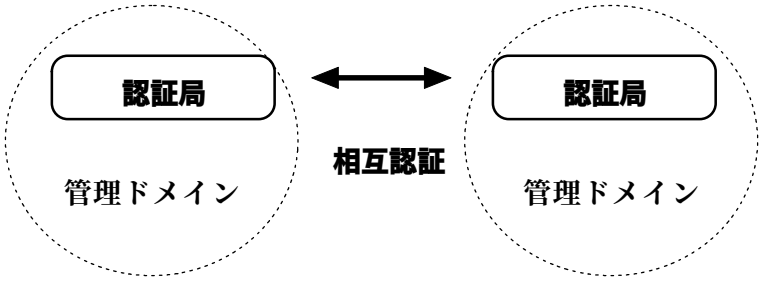


図１－３ 相互認証の位置付け

2.3.相互認証技術概要

この章では相互認証技術の基本となる方式を解説する。まず、ＩＥＴＦ等で最初にＲＳＡ社が公開した階層型認証技術を紹介し、これと対比して相互認証技術を解説する。

2.3.1.階層型認証技術

認証書を発行する認証局（ＣＡ）の信頼性を保証するために、階層型の認証局モデルでは上位の認証局が下位の認証局を認証するという方式になっている。最上位の認証局をルート認証局といい、このルート認証局のルート認証書は自らがルート認証用の署名鍵で署名した自己署名認証書である。ルート認証書の署名鍵自体の認証書は存在しない。このため、ルート認証局は信頼できる第三者機関（ＴＴＰ）として社会的かつ技術的な信頼性が必要とされる。

相手公開鍵の認証(Authentication)を行うための手段として認証書(Certificate)が提供される。この認証は認証経路(Certificate Path)に従って、最上位のルート認証書まで確認される。

階層型モデルでは、最上位のルート認証局はシステムで唯一の認証局である。相手認証書の検証はルート認証書まで正しく確認されれば認証経路の確認が完了する。

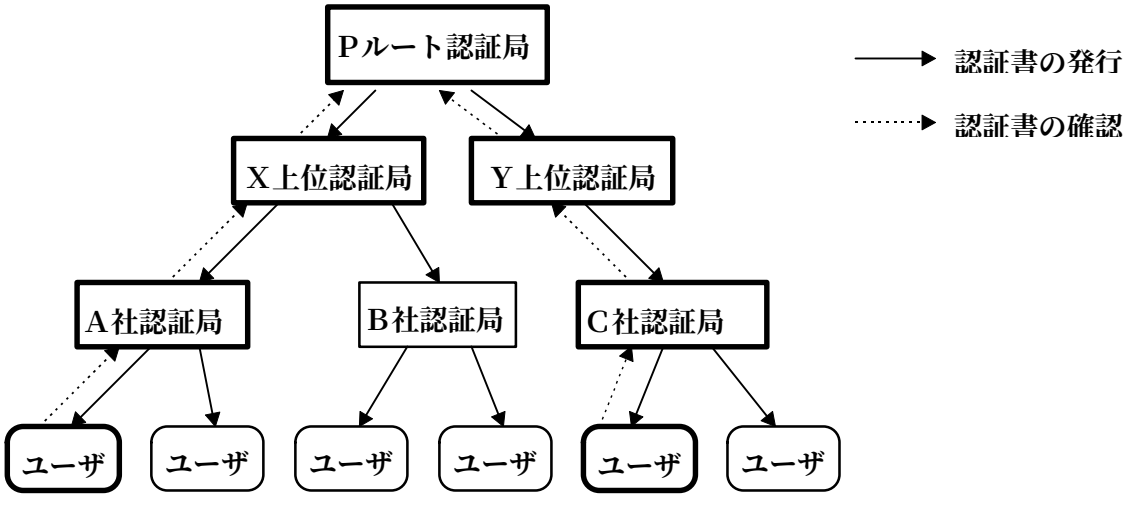


図 １－４ 階層型モデルの認証経路

2.3.2.相互認証技術

階層型モデルにおけるシステムとしてつぎの２つが前提となっている。

① 最上位の認証局であるルート認証局はシステムで唯一。

② 認証経路は最上位のルート認証書まで確認。

システムによってはこの前提を実現することが困難なことがある。

① ルート認証局が同様のサービスを提供するシステムの中で唯一ではなく、複数存在する場合。

② 認証局の階層が多段になり、認証経路の確認に伴う性能上の問題が無視できない場合。

サーバシステムがクライアントの認証を行う場合で、処理すべき認証書の数が性能上無視できない場合など。（最短経路が必要とされるシステム）

相互認証における認証経路の確認は単体で実現されるシステムではなく､階層型の認証経路の確認と相互認証による認証経路の確認とが混在する形態が現実的である｡この形態を前提とした相互認証による認証経路の確認方法はつぎの通りである。

【方式１】**相互認証経路優先方式**

この方式は最短経路が必要とされるシステムに向いている。相手公開鍵の認証書を発行した認証局の認証書をすべて検証することなく、自公開鍵の認証書を発行した認証局の相互認証書の発行者が相手公開鍵の認証局認証書の発行者に含まれる場合には、認証経路の確認が完了とみなす。

【方式２】**階層型認証経路優先方式**

この方式は最短経路の確認が必要とされるシステムには向いていないが、ルート認証局がシステムで唯一でない場合の認証経路の解決手段として適用される。

まず階層型の認証経路に従ってルート認証書までを確認するが、ルート認証書が自システムのルート認証書と異なる場合、認証局の相互認証書を検証する。相互認証経路によりその上位認証局が階層型のルートに結合している場合には、認証経路の確認が完了したとみなす。

(1) 相互認証経路優先方式

この方式では認証局の相互認証書が事前にユーザに配布されている事を前提としている。

何等かの手段によって相手の公開鍵認証書を入手した時､相手認証書の発行者名を確認する｡事前入手した相互認証書の所有者名を検査して発行者名と一致する所有者名があるかを確認する｡一致

する場合には、その時点で認証経路の確認を完了する。

簡便な方法といえるが、相互の信頼性を前提としているので、信頼性がそこなわれないような事前検討が必要である。

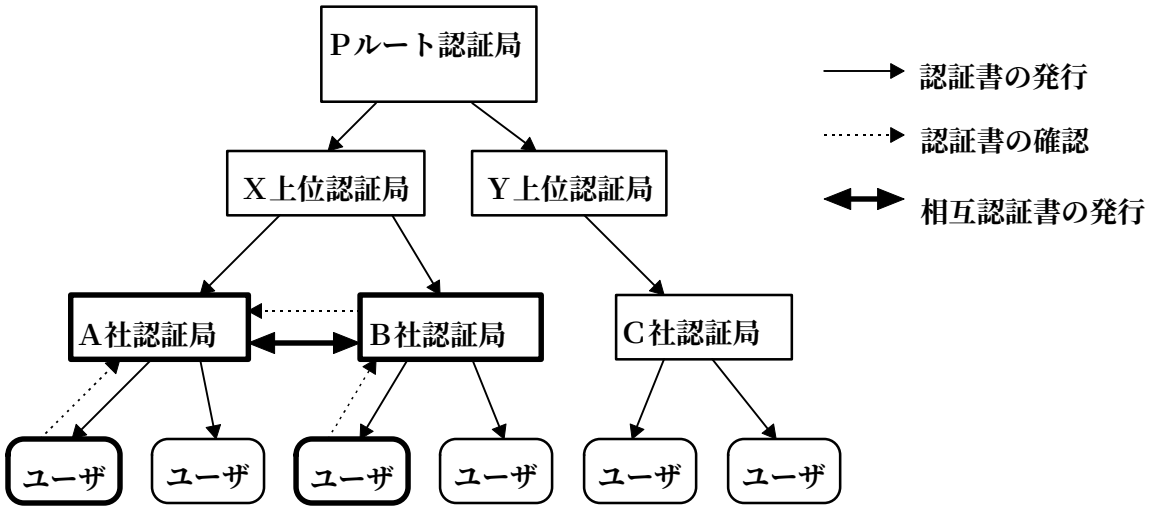


図 1－5 最短経路による相互認証経路

(2) 階層型認証経路優先方式

この方式は、異なるルート認証局を持つ場合のシステムに適用される。

まず階層型の認証経路に従ってルート認証局まで検証する。この時にルート認証書が自分のルート認証書と異なる場合には、ルート認証書の相互認証書を検証する。このルート相互認証書は何等かの手段によって入手できるものとする。ルート相互認証書の確認手段としてつぎの３通りがある。

① 順方向相互認証書による確認

この方式は相手のルート認証書に付随する順方向ルート相互認証書を検証する。順方向ルート相互認証書の発行者名が自ルート認証局名と一致する場合には、認証経路の確認を完了する。

② 逆方向相互認証書による確認

この方式は自ルート認証書に付随する逆方向ルート相互認証書を検証する。逆方向ルート相互認証書の所有者名が相手ルート認証局名と一致する場合は、認証経路の確認を完了する。

③ 両方相互認証書による確認

この方式は①及び②の確認を両方とも行ってから認証経路の確認を完了する方式である。

より厳密な相互認証を期待する場合に適用される。

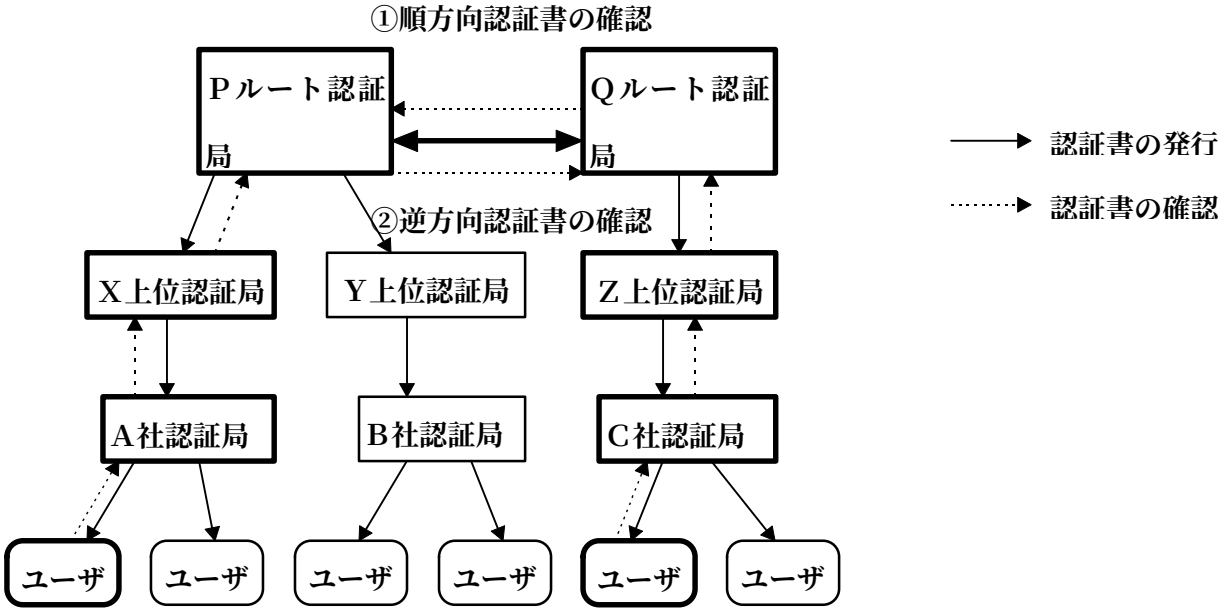


図１－６　異なるルート認証局経由の相互認証経路

2.3.3.相互認証の事例

相互認証の実際の事例をここでは紹介する。主に、米国の Natonal Institute of Standard and Technology（NIST）が開発した米国連邦政府公開鍵基盤技術仕様（Federal Public Key Infrastructure (PKI) Technical Specification: Part D- Interoperability Profiles）から引用している。

(1) 米国連邦政府の認証局構成

最上位のルート認証局をＰＡＡ（Policy Approving Authority）、第二レベルの認証局をＰＣＡ（Policy Creation Authority）、第三レベルの認証局をＣＡ（Certification Autority）という。さらに組織対応の登録局としてＯＲＡ（Organizational Registration Autority）がある。これらの構成図を図１－８に示す。

(2) インターネットの認証局構成

最上位のルート認証局をＩＰＲＡ（Internet Policy Registtration Autority）、第二レベルの認証局をＰＣＡ（Policy Certification Autority）、第三レベルの認証局をＣＡ（Certification Autority）という。さらにオプションとして登録局（Registration Autority）がある。

これらの構成図を図１－９に示す。

(3) 米国連邦政府の認証局とインターネット認証局の相互認証

ＰＡＡとＩＰＲＡまたはＰＡＡとＰＣＡのいずれかの相互認証形態が考えられている。

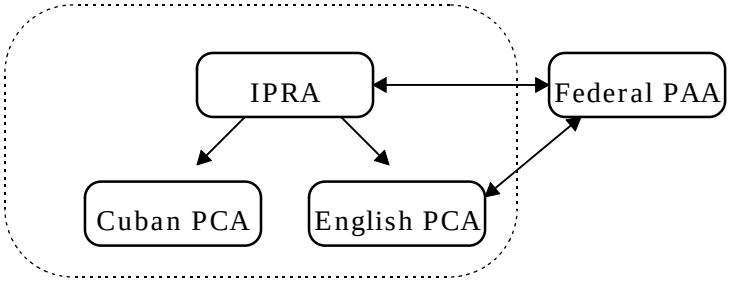


図１－７ 連邦政府認証局とインターネット認証局との相互認証経路

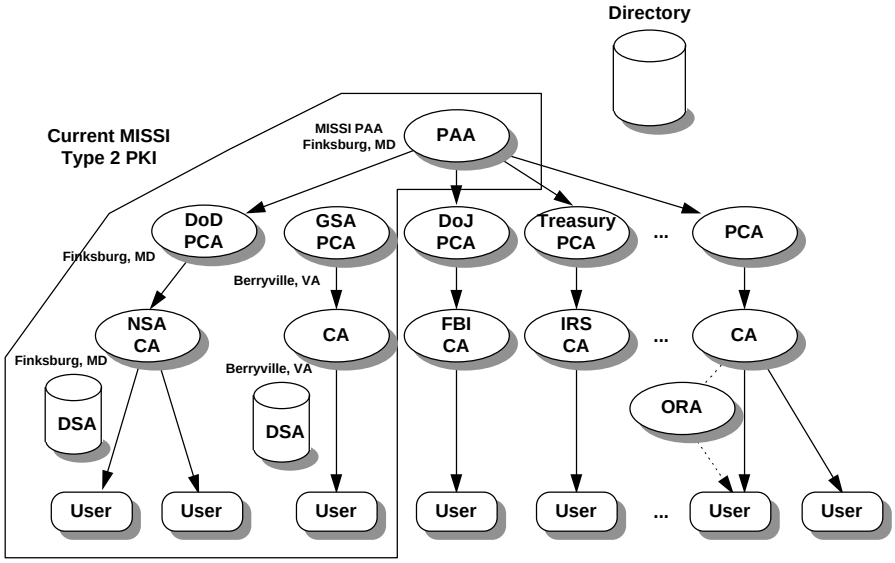


図１－８ 米国連邦政府認証局構成--- Federal PKI Architecture

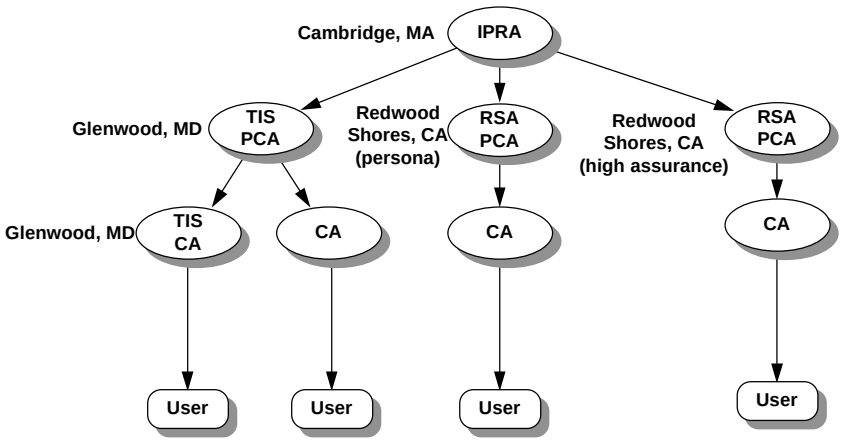


図１－９ インターネット認証局構成 -- Internet PKI Structure

(4) カナダ連邦政府の認証局構成と米国政府認証局との相互認証

このシステムはＤＣＥＫＭＳ（Designated/Electronic Commerce Canadian Electronic Key Management System）と呼び、レベル０からレベル４までの階層型の認証局構成となっている。最上位のレベル０認証局をＣＣＦ（Canadian Central Facility）、レベル１以下の認証局をＣＭＡ（Certificate Management Autority）という。さらにオプションとして登録局ＬＲＡ（Local Registration Autority）がある。

ＰＡＡとＣＣＦとの相互認証形態が考えられている。

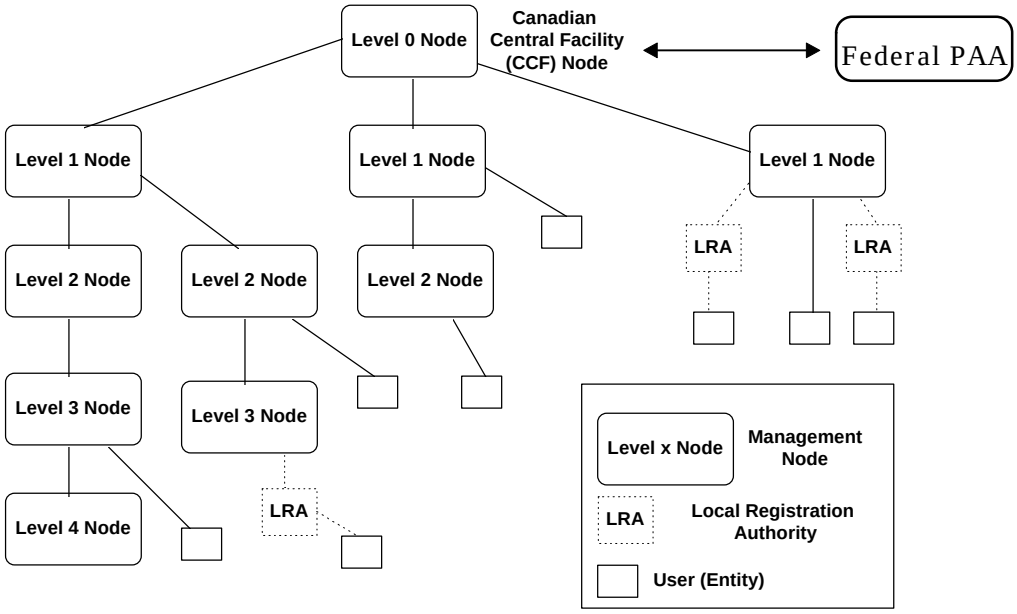


図１－１０ カナダ連邦政府の認証局構成 -- DCEKMS Structure

(5) 米国郵政サービスの認証局構成と米国政府認証局との相互認証

郵政サービスでは単一の認証局構成である。ＣＡは直接ユーザと接続し認証書を発行する。

ＰＡＡは郵政サービスのＣＡをＰＣＡの一つとみなして相互認証する。

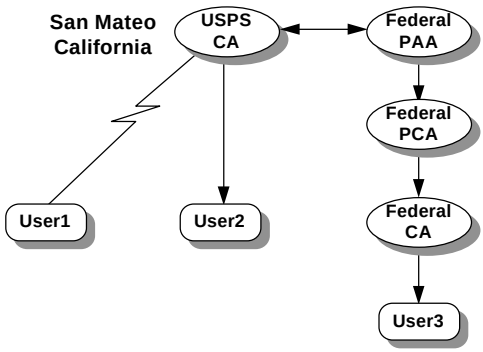


図１－１１ 米国連邦政府認証局とカナダ連邦政府認証局との相互認証経路

2.4.相互認証書の形式

ディレクトリィに保持される認証書の形式はX.509でつぎの３つの属性に分類されている。

（１） ユーザ認証書　UserCertificate

（２） 認証局認証書　CACertificate

（３） 相互認証書　CrossCertificatePair

この章で解説する相互認証書はＸ.５０９で定義する相互認証書（Cross-Certificate-Pairs）の属性を前提としている。[9] 順方向の相互認証と逆方向の相互認証とが定義されている。順方向とは相手側の認証局から自側の認証局に対する相互認証をいう。 逆方向とは自側から相手側に対する相互認証をいう。

```
crossCertificatePairATTRIBUTE   ::=     {
        WITH SYNTAX         CertificatePair
        ID                  id-at-crossCertificatePair }

CertificatePair             ::=     SEQUENCE {
        forward     [0]     Certificate OPTIONAL、
        reverse     [1]     Certificate OPTIONAL
                            -- at least one of the pair shall be present -- }
```

この両方向の相互認証書は少なくともいずれか一方のみ存在すればよいのであるが、 どの方向の相互認証を行うかは各認証局のポリシーによる。

認証局が他の認証局を相互認証する範囲は認証局相互に別途協議される。

各認証局の「認証実施規定(CPS)」と認証ポリシーを双方が事前審査する。各認証局はそのユーザによる認証経路確認に対する適当な制限を決定する。双方の合意により、各認証局は相互認証書を交換し、対になる認証書を作成する。各相互認証書をユーザが入手するための方法については各認証局の方針による。

2.4.1.相互認証書の基本構成

(1) 他の認証書と同様に相互認証書においてもＸ.５０９ Ｖ３ 認証書を基本構成とする。

基本構成上の必要最小限の実装規約については1.４.３ に規定する。

[9] Refer to ISO/IEC 9594-8 Clause8 Obtaining a user's public key

表１－１ 認証書の基本構成

基本部	**バージョン番号** Version	version3 = 2
	シリアル番号 Serial Number	認証局ごとの認証書番号
	発行者署名アルゴリズム Signature Algorithm	別表１を参照
	発行者名 Issuer Distinguished Name	X.500 識別名(別表２参照)
	有効期間 Varidity Period	開始年月日と終了年月日
	所有者名 Subject Distinguished Name	X.500 識別名(別表２参照)
	所有者公開鍵 Subject Public Key Info.	アルゴリズム識別と公開鍵
V2 拡張部	**発行者特定識別子** Issuer Unique Identifier	これらの識別子を使用しない事を推奨する。(有効期間を越えて再使用するための識別子である。)
	所有者特定識別子 Subject Unique Identifier	
V3 拡張部	----表４－２に示す----	
発行者署名		

(2) Ｘ.５０９で定義する認証書の定義はつぎの通りである。[10]

Ｖ３拡張部の表記については、ＮＩＳＴの資料[11] を参照している。

```
Certificate             ::=   SIGNED { SEQUENCE {
        version                 [0]     Version DEFAULT v1、
        serialNumber            CertificateSerialNumber、
        signature                       AlgorithmIdentifier、
        issuer                  Name、
        validity                        Validity、
        subject                         Name、
        subjectPublicKeyInfo            SubjectPublicKeyInfo、
        issuerUniqueIdentifier  [1]     IMPLICIT UniqueIdentifier OPTIONAL、
                                        -- if present、 version must be v2
        subjectUniqueIdentifier [2]     IMPLICIT UniqueIdentifier OPTIONAL
                                        ---if present、 version must be v1 or v2--
        extensions              [3]     Extensions Optional
                                        --if present、 version must be v3--}  }

Version                 ::=   INTEGER { v1(0)、 v2(1) }

CertificateSerialNumber ::=   INTEGER

AlgorithmIdentifier     ::=   SEQUENCE {
        algorithm               ALGORITHM.&id ({SupportedAlgorithms})、
        parameters              ALGORITHM.&Type ({SupportedAlgorithms}{ @algorithm}) OPTIONAL }
--      Definition of the following information object set is deferred、 perhaps to standardized
--      profiles or to protocol implementation conformance statements. The set is required to
--      specify a table constraint on the parameterscomponent of AlgorithmIdentifier.
--      SupportedAlgorithms     ALGORITHM  ::=     { ... | ... }

Validity                ::=   SEQUENCE {
        notBefore       UTCTime、
        notAfter        UTCTime }

SubjectPublicKeyInfo    ::=   SEQUENCE {
        algorithm               AlgorithmIdentifier、
        subjectPublicKey        BIT STRING }

Extensions              ::=   SEQUENCE OF Extension

Extension               ::=   SEQUENCE {
        extnId          EXTENSION.&id ({ExtensionSet})、
```

[10] Refer to ISO/IEC 9594-8:1993 (E)、 Amendment 1 to ISO/IEC 9594-8:1995 (E).
[11] NIST Minimum Interoperability Specification for PKI Components:1996-12-2

critical **BOOLEAN DEFAULT FALSE、**
extnValue **OCTET STRING**
-- contains a DER encoding of a value of type &ExtnType for the
¤ extension object identified by extnId -
-

表１－１ 別表１ 署名アルゴリズム

区分	署名とハッシュの組み合わせ	出典	適用例
RSA	RSA with MD2	PKCS #1	IETF PKIX[1]、 SSL[2]
	RSA with MD5		SSL[2]
	RSA with SHA－1	FIPS 180-1	SET[3]、 NIST MISPC[4]
DSA	DSA with SHA－1	FIPS 186	IETF PKIX[1]、 NIST MISPC[4]

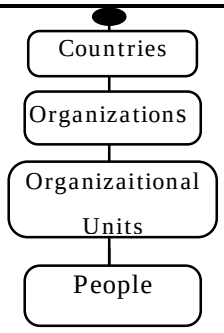
md2WithRSAEncryption OBJECT IDENTIFIER ::= {
iso(1) member-body(2) US(840) rsadsi(113549) pkcs(1)
pkcs-1(1) 2 }

md5WithRSAEncryption OBJECT IDENTIFIER ::= {
iso(1) member-body(2) US(840) rsadsi(113549) pkcs(1)
pkcs-1(1) 4 }

sha-1WithRSAEncryption OBJECT IDENTIFIER ::= {
iso(1) identified-organization(3) oiw(14) secsig(3)
algorithm(2) 29 }

dsaWithSHA-1 OBJECT IDENTIFIER ::= {
iso(1) identified-organization(3) oiw(14) secsig(3)
algorithm(2) 27 }

表１－１ 別表２ X.５００識別名[12]

DIT	RDN	認証局名への適用例		
		USA Federal	SSL	SET
	root			
	C＝	US	US	ISO 3166 国名コード
	O＝	U.S.Federal Govt.	Verisign	認証組織名
	OU＝	DoD NSA		
	CN＝	MISSI	Verisign Class1root	ユニークな CAのID

DIT: Directory Information Tree 、 RDN: Relative Distinguished Name

X.520 で定義する名前の属性はつぎの通りである。

commonName ATTRIBUTE WITH ATTRIBUTE-SYNTAX
caseIgnoreStringSyntax(SIZE(1..ub-common-name)
::= { attributeType 3}
countryName ATTRIBUTE WITH ATTRIBUTE-SYNTAX
PrintableString(SIZE(2)) - IS 3166 codes only
MATCHES FOR EQUALITY SINGLE VALUE
::= {attributeType 6}
organizationName ATTRIBUTE WITH ATTRIBUTE-SYNTAX
caseIgnoreStringSyntax(SIZE(1..ub-organization-name)
::= {attributeType 10}

[12] Refer to X.501 Determination of distinguished name.

2.4.2.相互認証書の拡張構成

Ｖ３拡張部は拡張情報識別子とクリティカル識別子及び拡張情報の３つから各情報ごとに構成されている。クリティカル識別子の定義はX.509 V3での標準定義を示している。

クリティカル識別子はその情報を受信する側がその情報を認識すべき(YES)か無視(NO)してよいかを示している。その扱いを規定しないものは任意(YES/NO)としている。

本参考資料での実装規約については１.４.３に規定する。

表１－２ 認証書Ｖ３拡張部の構成

区分	V3拡張部	クリティカル	拡張詳細情報	属性
(1)	**認証局鍵識別** Autority Key Identifier	無視 (NO)	鍵識別情報	[0]OCTET STRING
			認証書発行者名	[1]General Names
			認証書シリアル番号	[2]Serial Number
	所有者鍵識別 Subject Key Identifier	無視 (NO)	鍵識別情報	OCTET STRING
	鍵種別 Key Usage	任意	鍵種別情報	BIT STRING
	秘密鍵使用期間 Private Key Usage Period	任意	使用開始/終了日	Generalized Time
	認証局ポリシー Certificate Policies	任意	ポリシー識別子	OBJECT IDENTIFIER
			ポリシー権限情報	OBJECT IDENTIFIER
	ポリシー関連付け Policy Mappings	無視	認証局ポリシー識別子	OBJECT IDENTIFIER
(2)	**所有者別名** Subject Alternative Name	任意	一般名	General Names
	発行者別名 Issuer Alternative Name	任意	一般名	General Names
	所有者ディレクトリィ属性 Subject Directory Attribute	無視	ディレクトリィ属性	Attribute
(3)	**基本制限** Basic Constraints	任意	認証局	BOOLEAN
			認証経路長制限	INTEGER
	名前制限 Name Constraints	任意	許容サブツリー	General Names
			除外サブツリー	BaseDistance
	ポリシー制限 Policy Constraints	任意	ポリシーセット	OBJECT IDENTIFIER
			明示的ポリシー要求	INTEGER
			ポリシー関連付け禁止	INTEGER
(4)	**ＣＲＬ配布元** CRL Distribution Points	任意	配布元	Name
			理由	Flags
			ＣＲＬ発行者	General Names

一般名はX.509V3で定義されておりつぎの通りである。

```
GeneralName ::= CHOICE {
        otherName                       [0]     INSTANCE OF OTHER-NAME、
        rfc822Name                      [1]     IA5String、
        dNSName                 [2]     IA5String、
        x400Address                     [3]     ORAddress、
        directoryName           [4]     Name、
        ediPartyName            [5]     EDIPartyName、
        uniformResourceIdentifier       [6]     IA5String、
        iPAddress               [7]     OCTET STRING、
```

```
        registeredID                    [8]     OBJECT IDENTIFIER }
```

(1) 鍵及びポリシー情報 (key and policy informationy information)

つぎの６種類のパラメタが拡張定義されている。

ａ） **認証局鍵識別** (*Authority key identifier)*
ｂ） **所有者鍵識別** (*Subject key identifier*)
ｃ） **鍵種別** (*Key usage*)
ｄ） **秘密鍵使用期間** (*Private key usage period*)
ｅ） **認証局ポリシー** (*Certificate policies*)
ｆ） **ポリシー関連付け** (*Policy mappings*)

ａ） 認証局鍵識別

認証局の鍵更新が一定期間あるいは特別の状況において発生する。認証書の署名を確認するために使用する公開鍵の識別子である。

```
authorityKeyIdentifier EXTENSION ::= {
        SYNTAX              AuthorityKeyIdentifier
        IDENTIFIED BY       { id-ce 35 } }

AuthorityKeyIdentifier ::= SEQUENCE {
        keyIdentifier                           [0] KeyIdentifier
OPTIONAL、
        authorityCertIssuer             [1] GeneralNames            OPTIONAL、
        authorityCertSerialNumber       [2] CertificateSerialNumber OPTIONAL }
        ( WITH COMPONENTS               {...、 authorityCertIssuer PRESENT、
                                                authorityCertSerialNumber PRESENT} |
        WITH COMPONENTS {...、 authorityCertIssuer ABSENT、
                                                authorityCertSerialNumber ABSENT} )
KeyIdentifier ::= OCTET STRING
```

ｂ） 所有者鍵識別

通常、認証書の所有者は目的別に複数の認証書とそれぞれに対応する公開鍵を所有している。認証書の所有ユーザが目的別の公開鍵と認証書を識別するための所有者の公開鍵の識別子である。

```
subjectKeyIdentifier EXTENSION ::= {
        SYNTAX              SubjectKeyIdentifier
IDENTIFIED BY  { id-ce 14 } }

SubjectKeyIdentifier ::= KeyIdentifier
```

ｃ） 鍵種別

所有者の鍵更新も一定期間あるいは特別の状況において発生する。同一の鍵所有者の異なる公開鍵を識別するための所有者公開鍵の鍵種別である。

```
keyUsage EXTENSION ::= {
        SYNTAX          KeyUsage
        IDENTIFIED BY { id-ce 15 } }

KeyUsage ::= BIT STRING {

        digitalSignature        (0)、
        nonRepudiation          (1)、
        keyEncipherment         (2)、
```

```
        dataEncipherment        (3)、
        keyAgreement            (4)、
        keyCertSign             (5)、
        cRLSign                 (6) }
```

d） 秘密鍵使用期間

認証された公開鍵と対応する秘密鍵は公開鍵の有効期限とは異なる期間使用される。

このために秘密鍵の使用期間を認証書に表示可能とする要求がある。

```
privateKeyUsagePeriod EXTENSION ::= {
        SYNTAX          PrivateKeyUsagePeriod
        IDENTIFIED BY { id-ce 16 } }

PrivateKeyUsagePeriod ::= SEQUENCE {
        notBefore       [0]     GeneralizedTime OPTIONAL、
        notAfter[1]     GeneralizedTime OPTIONAL }
        ( WITH COMPONENTS   {...、 notBefore PRESENT} |
        WITH COMPONENTS     {...、 notAfter PRESENT} )
```

e） 認証局ポリシー

認証書は複数の認証ポリシーが適用される環境で使われるかもしれないので、認証局のポリシーを認証書で提供する事がある。

```
certificatePolicies EXTENSION ::= {
        SYNTAX          CertificatePoliciesSyntax
        IDENTIFIED BY { id-ce 32 } }

CertificatePoliciesSyntax ::= SEQUENCE SIZE (1..MAX) OF PolicyInformation

PolicyInformation ::= SEQUENCE {
        policyIdentifier    CertPolicyId、
        policyQualifiers    SEQUENCE SIZE (1..MAX) OF
                PolicyQualifierInfo OPTIONAL }

CertPolicyId ::= OBJECT IDENTIFIER

PolicyQualifierInfo ::= SEQUENCE {
        policyQualifierId CERT-POLICY-QUALIFIER.&id
                                ({SupportedPolicyQualifiers})、
        qualifier         CERT-POLICY-QUALIFIER.&Qualifier
                                ({SupportedPolicyQualifiers}{@policyQualifierId})
                                OPTIONAL }

SupportedPolicyQualifiers CERT-POLICY-QUALIFIER ::= { ... }

CERT-POLICY-QUALIFIER ::= CLASS {
        &id             OBJECT IDENTIFIER UNIQUE、
        &Qualifier      OPTIONAL }
WITH SYNTAX {
        POLICY-QUALIFIER-ID   &id
        [QUALIFIER-TYPE       &Qualifier] }
```

f） ポリシー関連付け

ある組織から他の組織に対して相互認証を行う時、両者の組織のポリシーの幾つかは同一のポリシーであると想定されている。このようなポリシーの対応付けをポリシー関連付けという。

```
policyMappings EXTENSION ::= {
        SYNTAX          PolicyMappingsSyntax
        IDENTIFIED BY { id-ce 33 } }

PolicyMappingsSyntax ::= SEQUENCE SIZE (1..MAX) OF SEQUENCE {
        issuerDomainPolicy              CertPolicyId、
        subjectDomainPolicy     CertPolicyId }
```

(2) 認証書所有者属性及び認証書発行者属性

(certificate subject and certificate issuer attributes)

つぎの３種類のパラメタが拡張定義されている。

ａ） **所有者別名** (*Subject alternative name)*
ｂ） **発行者別名** (*Issuer alternative name*)
ｃ） **所有者ディレクトリィ属性** (*Subject directory attributes*)

ａ） 所有者別名及び発行者別名

認証書は様々な形式を持つアプリケーションによって使用できなければならない。

インターネットの電子メール名、インターネットドメイン名、X.400 発信/受信者アドレス、及びＥＤＩ組織名などが含まれる。このため認証書において複数の名称形式を表現できることが必要である。

```
subjectAltName EXTENSION ::= {
        SYNTAX          GeneralNames
        IDENTIFIED BY { id-ce 17 } }

issuerAltName EXTENSION ::= {
        SYNTAX          GeneralNames
        IDENTIFIED BY { id-ce 18 } }

GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName

GeneralName ::= CHOICE {
        otherName                       [0]     INSTANCE OF OTHER-NAME、
        rfc822Name                      [1]     IA5String、
        dNSName                         [2]     IA5String、
        x400Address                     [3]     ORAddress、
        directoryName           [4]     Name、
        ediPartyName            [5]     EDIPartyName、
        uniformResourceIdentifier       [6]     IA5String、
        iPAddress                       [7]     OCTET STRING、
        registeredID                    [8]     OBJECT IDENTIFIER }

OTHER-NAME ::= TYPE-IDENTIFIER

EDIPartyName ::= SEQUENCE {
        nameAssigner                    [0]     DirectoryString {ub-name} OPTIONAL、
        partyName                       [1]     DirectoryString {ub-name} }
```

(b) 所有者ディレクトリィ属性

認証書ユーザはある所有者に関して、その所有者が実際に意図したその人物であるという事を確認するためにある識別情報を安全に知る必要があると思われる。(郵便の住所、企業内

の所属、写真イメージなど）その様な情報はディレクトリィ属性として表現するのが便利であると思われるが、 それらの属性は識別名の一部としては不要である。

このような識別名を越える付加的なディレクトリィ属性を運ぶための情報が認証書に必要である。

```
subjectDirectoryAttributes EXTENSION ::= {
        SYNTAX          AttributesSyntax
        IDENTIFIED BY { id-ce 9 } }

AttributesSyntax ::= SEQUENCE SIZE (1..MAX) OF Attribute
```

(3) 認証経路の制限 (certification pathtion path constraints)

X.509 における認証経路の制限はつぎのような考え方に基づいている。

・たとえば信頼性の低いＣＡが不適当な名前を持つ認証書を発行したためにシステムの信用がそこなわれていないかを認証書のユーザがチェックできるようにするために、ＣＡが認証経路上の制限を規定することがある。

・複数の認証ポリシーが容認されている環境で認証経路を指定することがある。ＣＡは他のドメインのＣＡを信用するかどうかを規定する。複数のポリシードメイン間の連携をサポートする場合に使用する。

・単一の組織の場合には階層モデルの認証経路でシステムが閉じているが、 複数の組織が相互接続された環境では認証経路の柔軟性が必要になる。

・認証ポリシーの異なるＣＡとの相互接続を拒否する場合のためにポリシーマッピングの使用を禁止することがある。

つぎの３種類のパラメタが拡張定義されている。

ａ）**基本制限**(*Basic constraints*)
ｂ）**名前制限**(*Name constraints*)
ｃ）**ポリシー制限**(*Policy constraints*)

ａ）基本制限

認証書の所有者がＣＡの場合に存在し、認証経路の長さ制限を規定する。

```
basicConstraints EXTENSION ::= {
        SYNTAX          BasicConstraintsSyntax
        IDENTIFIED BY { id-ce 19 } }

BasicConstraintsSyntax ::= SEQUENCE {
        cA                      BOOLEAN DEFAULT FALSE、
        pathLenConstraint       INTEGER (0..MAX) OPTIONAL }
```

ｂ）名前制限

ＣＡの認証書においてのみ使用される。認証書所有者の名前の長さを規定する。

```
nameConstraints EXTENSION ::= {
        SYNTAX          NameConstraintsSyntax
        IDENTIFIED BY { id-ce 30 } }

NameConstraintsSyntax ::= SEQUENCE {
        permittedSubtrees       [0]     GeneralSubtrees OPTIONAL、
        excludedSubtrees        [1]     GeneralSubtrees OPTIONAL }

GeneralSubtrees ::= SEQUENCE SIZE (1..MAX) OF GeneralSubtree

GeneralSubtree ::= SEQUENCE {
        base                    GeneralName、
        minimum                 [0]     BaseDistance DEFAULT 0、
        maximum                 [1]     BaseDistance OPTIONAL }

BaseDistance ::= INTEGER (0..MAX)
```

ｃ）ポリシー制限

明示的な認証ポリシーの表示及びポリシーマッピングの禁止の表示を規定する。

```
policyConstraints EXTENSION ::= {
        SYNTAX          PolicyConstraintsSyntax
        IDENTIFIED BY { id-ce 34 } }

PolicyConstraintsSyntax ::= SEQUENCE SIZE (1..MAX) OF SEQUENCE {
        policySet                       [0] CertPolicySet OPTIONAL、
        requireExplicitPolicy   [1] SkipCerts OPTIONAL、
        inhibitPolicyMapping            [2] SkipCerts OPTIONAL }

SkipCerts ::= INTEGER (0..MAX)

CertPolicySet ::= SEQUENCE SIZE (1..MAX) OF CertPolicyId
```

(4) ＣＲＬ配布元（CRL distribution points）

X.509 におけるＣＲＬの配布元に関する考え方はつぎの通りである。

リボケーションリストは数が多くなり扱いにくくなる可能性がある。このため部分的なＣＲＬを提供できる必要がある。

このようなＣＲＬは大規模サーバシステムではＣＲＬの更新により、前回のＣＲＬと現在の最新のＣＲＬとの差分情報として提供される。

このような差分ＣＲＬの配布元の指定に適用される。

```
cRLDistributionPoints EXTENSION ::= {
        SYNTAX          CRLDistPointsSyntax
        IDENTIFIED BY   { id-ce 31 } }

CRLDistPointsSyntax ::= SEQUENCE SIZE (1..MAX) OF DistributionPoint

DistributionPoint ::= SEQUENCE {
        distributionPoint       [0]     DistributionPointName OPTIONAL、
        reasons         [1]     ReasonFlags OPTIONAL、
        cRLIssuer               [2]     GeneralNames OPTIONAL }

DistributionPointName ::= CHOICE {
        fullName                        [0]             GeneralNames、

        nameRelativeToCRLIssuer [1]             RelativeDistinguishedName }
```

```
ReasonFlags ::= BIT STRING {
        unused                  (0)、
        keyCompromise           (1)、
        cACompromise            (2)、
        affiliationChanged      (3)、
        superseded              (4)、
        cessationOfOperation    (5)、
        certificateHold         (6) }
```

2.4.3.相互認証書の実装規約

Ｘ.５０９Ｖ３で定義されている認証書の形式を相互認証書に適用する場合の各情報の実装条件について規定する。既存の実装規約との比較を合わせて引用している。

(1) 基本部の実装規約

基本部の情報は相互接続性を重視して相互認証書においてすべて「使用する」とする。

Ｖ２拡張部の情報は「使用しない」とする。

表１－３ 相互認証実装規約（基本部及びＶ２拡張部）

○：使用する、 △ :使用任意 × ：使用しない

区分	パラメタ	ＰＫＩＸ実装規約[13]	ＳＥＴ実装規約[14]	定義	相互認証実装規約
基本部	バージョン番号（version）	○	○(V3)	V3 を指定	○(V3)
	シリアル番号（serial Number）	○	○	CA が指定	○
	署名情報（signature） ・AlgorithmIdentifier	○	○	署名アルゴリズム	○
	発行者名（issuer）	△	○	応答 CA 名	○
	有効期間（validity） ・notBefore ・notAfter	○	○	有効期間	○
	所有者名（subject）	△	○	要求 CA 名	○
	所有者公開鍵情報（subject Public Key Info） ・AlgorithmIdentifier	○	○	要求 CA の公開鍵	○
V2 拡張部	issuer Unique Identifier	×	×		×
	subject Unique Identifier	×	×		×

【解説】相互の信頼に基づく任意の認証局間の相互認証書の基本部の形式は、ＣＡ認証書と同じ形式を適用する。ＣＡ認証書の発行者及び署名は階層モデルでは上位ＣＡであるが、相互認証の場合には、異なるドメインのＣＡが発行者になる。

認証局の公開鍵とそれに対応する認証書は４種類ある。

(1) メッセージ署名公開鍵とメッセージ署名公開鍵認証書

(2) セッション鍵暗号化公開鍵と鍵暗号化公開鍵認証書

(3) 認証書署名公開鍵と認証書署名公開鍵認証書

(4) ＣＲＬ署名公開鍵とＣＲＬ署名公開鍵認証書

---

[13] Refer to IETF Draft “Intenet Public Key Infrastructure Part1: X.509 Certificate and CRL Profile”

[14] Refer to “Secure Electronic Transaction (SET) Specification Book2: Programmer’s Guide”

相互認証書ではこの中の、 少なくとも（3）認証書署名公開鍵認証書 を作成する。

その他の相互認証書の要否は、 相互認証書を相互に発行するための手順及びＣＲＬの発行ポリシーに依存する。

(2) 拡張部の実装規約

相互認証書のＶ３拡張部の情報は相互接続に関連するポリシー情報及び認証経路の制限情報に関しては「使用する」と規定する。その他の情報に関しては実装任意とする。

クリティカルの実装に関してはＮＩＳＴの実装規約を参考にしている。

表１－４ 相互認証実装規約（Ｖ３拡張部）

○ : 使用する、 △ :使用任意 × : 使用しない ✓：規定なし

C=: クリティカル、Y=認識、N=無視

区分	パラメタ	クリティカル	ＮＩＳＴ実装規約[15]	ＳＥＴ実装規約	相互認証実装規約
鍵及びポリシー情報	認証局鍵識別 Authority KeyIdentifier	無視	○	○	○(C=N)
	---鍵識別情報 KeyIdentifier		○	×	○
	---認証書発行者名 authorityCertIssuer		○	○	○
	---シリアル番号 authorityCertSerialNumber		○	○	○
	所有者鍵識別 Subject Key Identifier	無視	○	✓	△
	鍵種別 Key Usage	任意	○(C=Y)	○	△(C=Y)
	秘密鍵使用期間 Private Key Usage Period	任意	○(C=N)	○	△(C=N)
	認証局ポリシーCertificate Policies	任意	○(C=N)	○	○(C=N)
	---ポリシー識別 policyIdentifier		○	○	○
	---ポリシー権限 policyQualifiers		○	○	○
	ポリシー関連付け Policy Mappings	無視	○	✓	○(C=N)
認証書所有者属性発行者属性	所有者別名 Subject Alternative Name	任意	○(C=N)	△	○(C=N)
	発行者別名 Issuer Alternative Name	任意	○(C=N)	△	○(C=N)
	所有者ディレクトリィ属性 SubjectDirectoryAttributes	無視	○(C=N)	✓	△
認証経路の制限	基本制限 Basic Constraints	任意	○(C=Y)	○	○(C=Y)
	---認証局 cA		○	○	○
	---認証経路制限 pathLenConstraint		○	○	○
	名前制限 Name Constraints	任意	○(C=Y)	✓	○(C=Y)
	---許容サブツリーpermittedSubtree		○	✓	○
	---除外サブツリーexcludeSubtree		○	✓	○
	ポリシー制限 Policy Constraints	任意	○(C=Y)	✓	○(C=Y)
	---ポリシーセット policySet		○	✓	○
	---明示的ポリシー要求 requireExplicitPolicy		○	✓	○
	---ポリシー関連付禁止 inhibitPolicyMapping		○	✓	○
ＣＲＬ識別	ＣＲＬ配布元 CRL Distribution Points	任意	○(C=N)	✓	△(C=N)
	---配布元 distributionPoint		○	✓	△
	---理由 reasons		○	✓	△
	---ＣＲＬ発行者 cRLIssuer		○	✓	△

[15] Refer to NIST “Minimum Interoperability Specification for PKI Components”

2.5.相互認証プロトコル

この章では相互認証技術における２つの方式に対応する相互認証手順を中心に解説する。

これらの２つの方式の概要は 1.3.2 に解説している。

認証局が他の認証局を相互認証するかどうかの事前審査に関しては、この章では対象としない。

２．４章にこの事前審査に関する概要を解説している。相互認証手順として次の３つについて解説する。

① 相互認証書を認証局間で交換する手順

② 相互認証書をユーザに配布する手順

③ 相互認証書の転送形式

ユーザ認証書に基づく認証手順をアプリケーションが実行する時の相互認証書の利用手順については共通技術編では対象としない。個別技術編 に代表的な手順を解説する。

2.5.1.相互認証書の交換手順

相互認証書を認証局間で交換するための手順を解説する。

ユーザと認証局間とでユーザ認証書の発行を行う認証局では、 発行手順を認証局間に応用することで相互認証書の交換を実現することが可能となる。ここでは、各認証書発行手順に依存しない認証局間に共通の相互認証書交換手順を紹介する。

(1) ＩＥＴＦのＰＫＩＸ相互認証交換手順[16]

ＰＫＩ管理機能の付加機能の一つとして相互認証機能を定義している。

２方向要求応答型の相互認証手順として定義されている。相互認証を行う任意の認証局の一方を要求ＣＡ、他方を応答ＣＡという。

① まず応答ＣＡが認証開始コードを生成し、これを応答ＣＡから要求ＣＡに対してインターネット以外の別手段で渡す。

② 要求ＣＡは認証開始コードを入力してオンライン手順の開始を起動する。

③ 双方のＣＡは認証開始コードに基づいて対称鍵を生成し、この対称鍵によりオンラインで転送されるすべてのメッセージ認証コード（ＭＡＣ）を生成する。

[16] Refer to IETF draft “Internet Public Key Infrastructure Part3: Certificate Management Protocols”(96/12)

④ 要求ＣＡはランダム値を生成して相互認証要求 CrossReq を転送する。

⑤ 応答ＣＡはプロトコルバージョンをチェックしてから要求ランダム値を保存し、応答ランダム値を生成しさらにＭＡＣを確認する。

⑥ 応答ＣＡは要求ＣＡの公開鍵を含み応答ＣＡの署名秘密鍵で署名した新しい要求ＣＡ認証書を生成する。

⑦ 応答ＣＡは要求ＣＡ認証書を相互認証応答 CrossRep で転送する。

⑧ 要求ＣＡは相互認証応答を受信するとランダム値をチェックしＭＡＣを確認する。

⑨ 要求ＣＡは応答ＣＡの公開鍵を含み要求ＣＡの署名秘密鍵で署名した新しい応答ＣＡ認証書を生成する。

⑩ 要求ＣＡは応答ＣＡ認証書を公開鍵確認 PKIConfirm で転送する。

⑪ 応答ＣＡは公開鍵確認を受信するとランダム値をチェックし、応答ＣＡ認証書を保存し、ＭＡＣを確認する。要求ＣＡ認証書と応答ＣＡ認証書の両方を保存する。

⑫ 応答ＣＡは公開鍵確認 PKIConfirm を転送する。

⑬ 要求ＣＡは公開鍵確認を受信するとランダム値をチェックしＭＡＣを確認する。

要求ＣＡ認証書と応答ＣＡ認証書の両方を保存する。

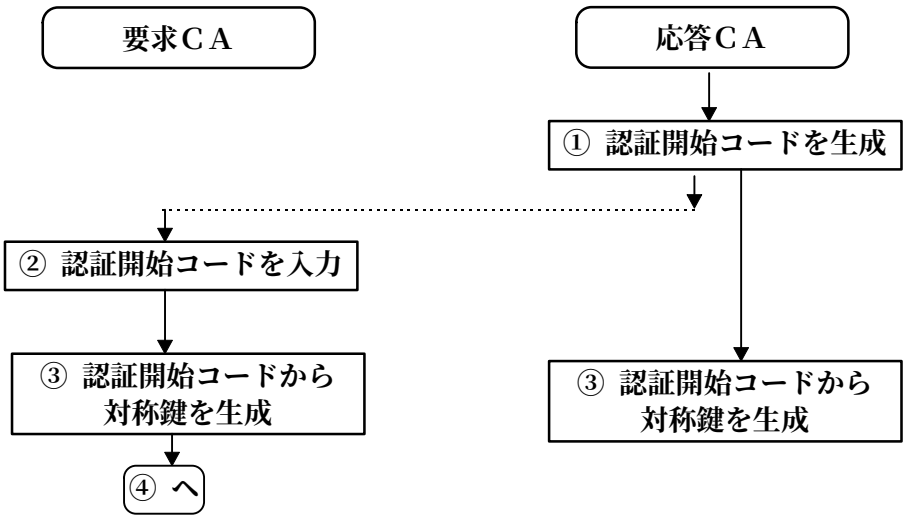


図１－１２　ＩＥＴＦの２方向相互認証手順（１）

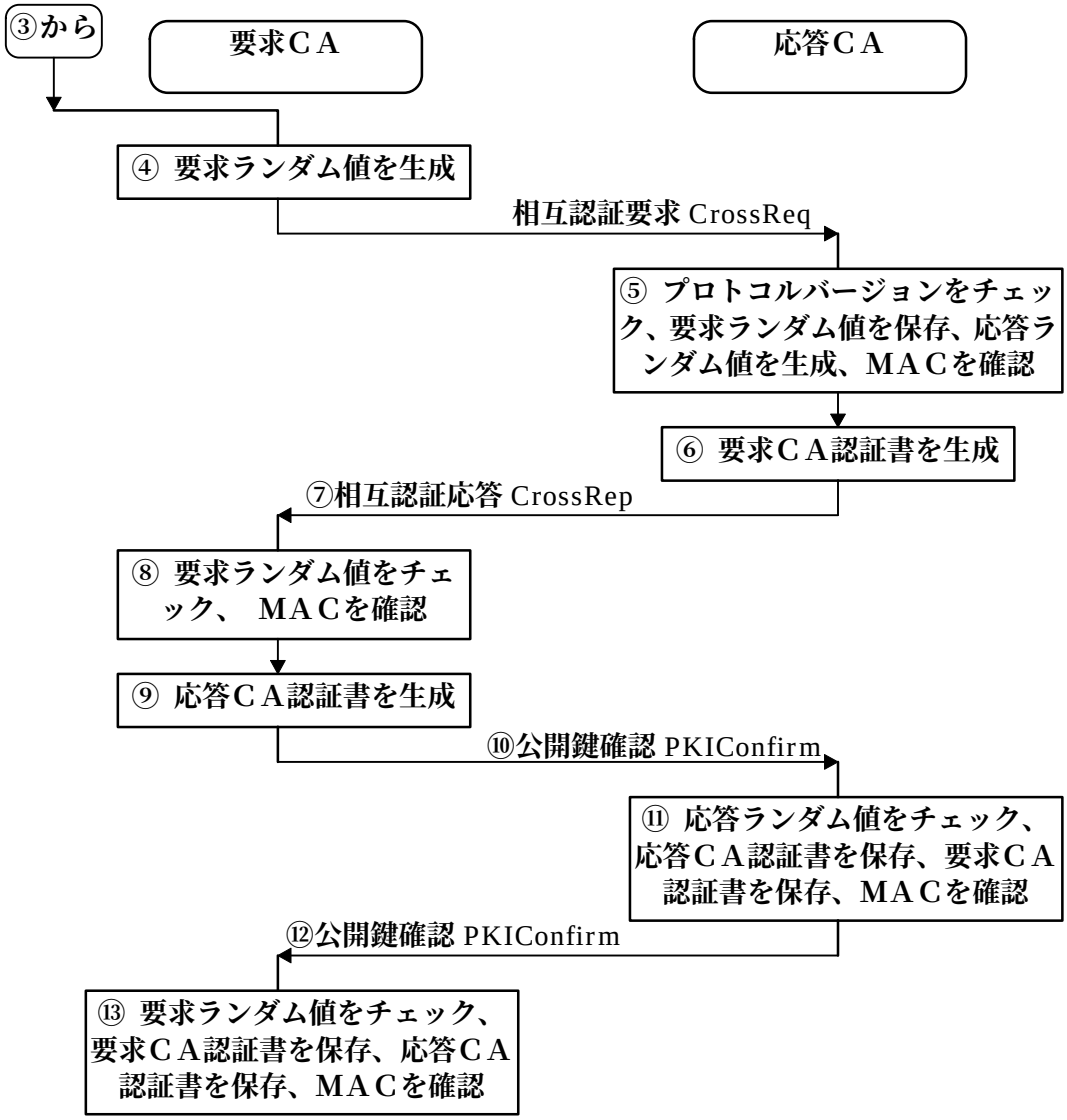


図１－１３　ＩＥＴＦの２方向相互認証手順（２）

(2) ＮＩＳＴの相互認証交換手順[17]

相互接続の事例を補足する形で相互認証交換手順に関するコメントが参考的に紹介されている。

任意の２つの認証局管理を行う適当な代表者が一同に介して協議することにより相互認証を実現する事を前提としている。つまり、すべてオフラインによる協議を前提とした手順である。

① 双方の公開鍵を交換する。

② 責任範囲（Liability）に関する契約を交わす。

③ 合意事項（Compromise notification）に関する契約を交わす。

④ 鍵の更新を含む回復手順に関する契約を交わす。

⑤ 互いの定期的な運用監査に関する契約を交わす。

⑥ 名前の制限長に関する契約を交わす。

⑦ ポリシーの関連付けに関する契約を交わす。互いに許容できるポリシーを規定する。

⑧ ディレクトリィの連携及びディレクトリィアクセス認証プロトコルを規定する。

⑨ 契約に署名し相互にコピーを提供する。

(3) 相互認証交換手順に関する考察

ＩＥＴＦの提案する相互認証交換手順は､双方の認証局の管理者が相互に信頼している前提の理想的な交換手順といえる｡双方が生成した相互認証書の内容が､事前に協議した合意事項と一致しているかの検証を双方の認証局の管理者が行うことは考慮の対象外になっている。

しかし､現実的には相互認証書の生成及び検証は､オンライン手順の一連の流れの中である程度の時間間隔を必要とすると考えられる｡この技術参考資料ではＩＥＴＦの手順をつぎの様に変更した手順を推奨する。（*1：応答ランダム値チェック、ＭＡＣ確認）

[17] Refer to NIST “Federal Public Key Infrastructure (PKI) Technical Specification : PartD - Interoperability Profiles
Appendix D Cross Certification Process”

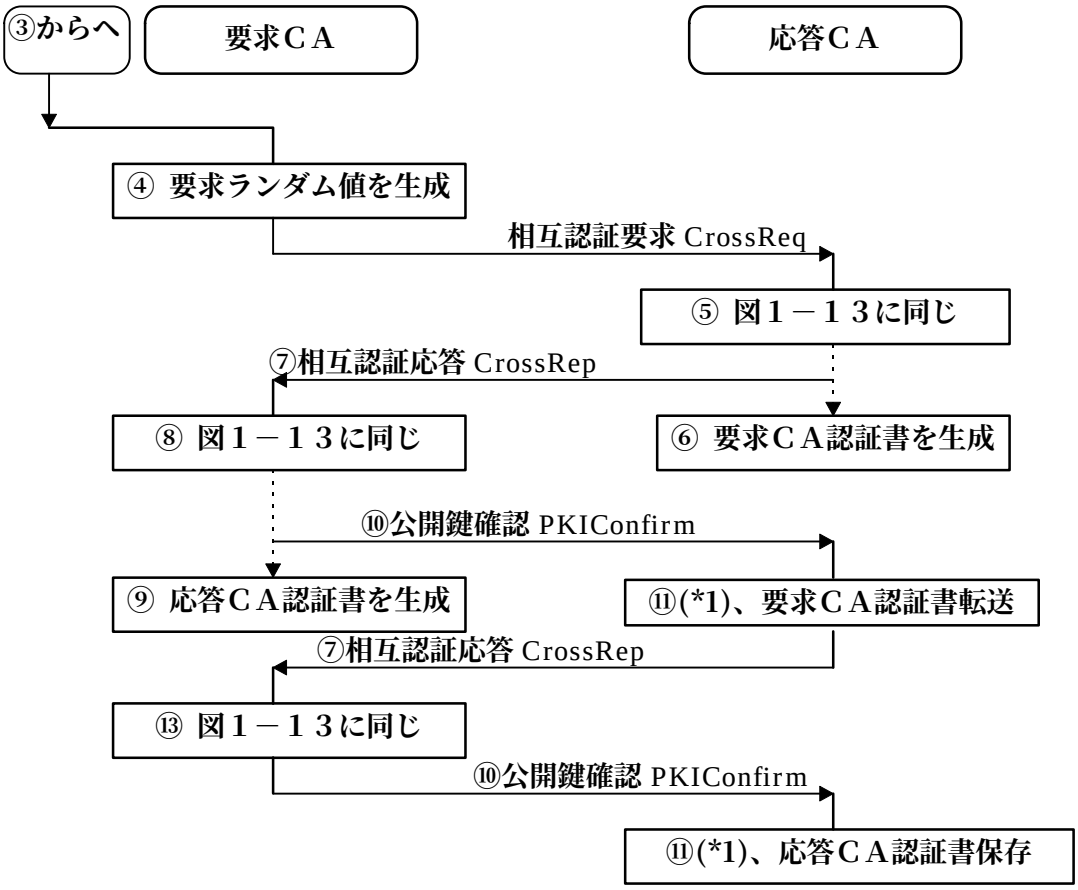


図１－１４ 相互認証交換手順

2.5.2.相互認証書の配布手順

相互認証書が認証局間で交換されることにより、以前には相互接続ができなかった別の認証ドメイン間での相互接続が可能となる。一般にこのような相互認証は、認証局間での契約によるため、相互接続ができる範囲の拡張に関して認証局のユーザが直接情報を入手することは困難と考えられる。相互認証書を認証経路の一つとして、ユーザが追加できる様にするためには、相互認証書を認証局がユーザに配布するための手順を規定しておく必要がある。

ＩＥＴＦ及びＮＩＳＴではこの方法として、相互認証書を公開ディレクトリィに登録するとしている。これも現実的には、その様なすべての認証局を対象とした公開ディレクトリィという理想的な環境を前提にはできない。この問題点の解決策を解説する。

ただし、ユーザの認証ソフトウェアが相互認証書の処理に対応していることが前提となる。

必要とするユーザに対しては、認証局が相互認証に対応しているユーザ側の認証ソフトウェアを配布できることが要求される。

(1) ユーザ認証書発行時の配布

ある認証局が既に他の認証局と相互認証書を交換している場合には、その認証局の新規ユーザに対するユーザ認証書を発行する時に、上位認証局認証書とともに相互認証書のペアを配布することが可能と考えられる。この場合には特別な配布手順はなく、認証書のコンテンツの追加となる。

(2) ソフトウェアによる配布

上に述べた様に、ユーザ側の認証ソフトウェアが相互認証に対応していない場合が考えられる。このような場合には認証局が相互認証に対応しているユーザ側の認証ソフトウェアを配布する際に、相互認証書のペアも同時に配布することが可能と考えられる。この場合の配布手順はオフラインによる方法（媒体に格納して配布）やオンラインによる方法（公開WWWサーバからダウンロード）など幾つかの方法がある。

(3) 電子メールによる配布

認証局が相互認証書を交換した後、ユーザ認証書を発行済みのユーザに対して電子メールで相互認証書の配布開始を通知する方法である。この時、電子メールには相互認証書をユーザが入手するためのサイトとセキュリティのためのチャレンジコードを通知する。

(4) ディレクトリィによる配布

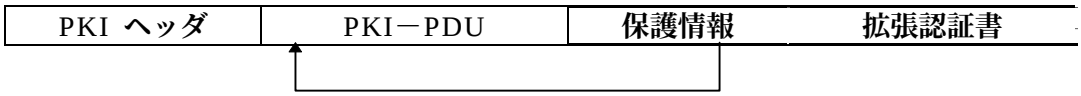
この方法は配布開始通知までは（３）と同じであるが、ユーザが入手するための手順としてＤＡＰ（Directory Access Protocol）を使用して公開ディレクトリィサーバから入手する方法である。ユーザ側はこれに対応するＬＤＡＰクライアントを実装していることが前提となる。

2.5.3.相互認証書の転送形式

相互認証プロトコルに対応する相互認証メッセージの転送形式について解説する。

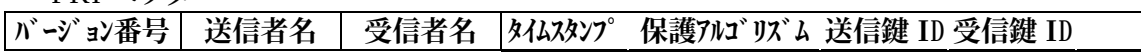
認証メッセージの転送形式としてＩＥＴＦが標準化提案しているＰＫＩメッセージの転送形式を紹介する。実際には各プロトコルごとに転送形式は異なる。個別編を参照して頂きたい。

(1) ＩＥＴＦのＰＫＩメッセージ転送形式




```
PKIMessage ::= SEQUENCE
                    header          PKIHeader、
                    body            PKIBody、
                    protection      [0] PKIProtection OPTIONAL、
                    extraCerts      [1] SEQUENCE OF Certificate OPTIONAL}
```

PKI ヘッダ




```
PKIHeader ::= SEQUENCE
            pvno                INTEGER     { ietf-version1 (0) }、
            sender              GeneralName、
            -- identifies the sender
            recipient           GeneralName、
            -- identifies the intended recipient
            messageTime         [0] GeneralizedTime         OPTIONAL、
            -- time of production of this message (used when sender
            -- believes that the transport will be "suitable"; i.e.、
            -- that the time will still be meaningful upon receipt)
            protectionAlg       [1] AlgorithmIdentifier     OPTIONAL、
            -- algorithm used for calculation of protection bits
            senderKID           [2] KeyIdentifier           OPTIONAL、
            recipKID            [3] KeyIdentifier           OPTIONAL、
            -- to identify specific keys used for protection
            transactionID       [4] OCTET STRING            OPTIONAL、
            -- identifies the transaction、 i.e. this will be the same in
            -- corresponding request、 response and confirmation messages
            senderNonce         [5] OCTET STRING            OPTIONAL、
            recipNonce          [6] OCTET STRING            OPTIONAL、
            -- nonces used to provide replay protection、 senderNonce
            -- is inserted by the creator of this message; recipNonce
            -- is a nonce previously inserted in a related message by
```

```
                -- the intended recipient of this message
                freeText        [7] PKIFreeText             OPTIONAL
                -- this may be used to indicate context-specific
                -- instructions (this field is intended for human
                -- consumption) }

PKIFreeText ::= CHOICE
                iA5String   [0] IA5String、
                bMPString   [1] BMPString }

PKIProtection ::= BIT STRING
```

(2) ＩＥＴＦのＰＫＩ-ＰＤＵ形式

CrossCertReq-PDU

PID	Cert 要求 ID	X.509 認証書テンプレート	署名ｱﾙｺﾞﾘｽﾞﾑ	POPO 署名	PKIｱｰｶｲﾌﾞｵﾌﾟｼｮﾝ

PKI 発行有無	PKI 発行方法	PKI 発行者名	旧発行者名	旧ｼﾘｱﾙ番号

CrossCertRep-PDU

PID	X.509 認証書	Cert 要求 ID	PKIｽﾃｰﾀｽ	PKI 認証失敗理由	X.509 認証書

暗号化 X.509 認証書	暗号化秘密鍵	PKI 発行有無	PKI 発行方法	PKI 発行者名

暗号化**

暗号化値	暗号化ｱﾙｺﾞﾘｽﾞﾑ	対称鍵ｱﾙｺﾞﾘｽﾞﾑ	対称鍵	ｱﾙｺﾞﾘｽﾞﾑ識別子

```
PKIBody ::= CHOICE {          -- message-specific body elements
                ir              [0]   InitReqContent、
                ip              [1]   InitRepContent、
                cr              [2]   CertReqContent、
                cp          [3]   CertRepContent、
                kur         [4]   KeyUpdReqContent、
                kup         [5]   KeyUpdRepContent、
                krr         [6]   KeyRecReqContent、
                krp         [7]   KeyRecRepContent、
                rr          [8]   RevReqContent、
                rp          [9]   RevRepContent、
                ccr         [10] CrossCertReqContent、
                ccp         [11] CrossCertRepContent、
                ckuann      [12] CAKeyUpdAnnContent、
                cann        [13] CertAnnContent、
                rann        [14] RevAnnContent、
                crlann      [15] CRLAnnContent、
                conf        [16] PKIConfirmContent、
                nested      [17] NestedMessageContent、
                infor       [18] PKIInfoReqContent、
                infop       [19] PKIInfoRepContent、
                error       [20] ErrorMsgContent }

CrossCertReqContent ::= CertReqContent
CertReqContent ::= FullCertTemplates

FullCertTemplates ::= SEQUENCE OF FullCertTemplate
```

```
FullCertTemplate ::= SEQUENCE {
        certReqId               INTEGER、
        -- to match this request with corresponding response
        -- (note:  must be unique over all FullCertReqs in this message)
        certTemplate            CertTemplate、
        popoSigningKey      [0] POPOSigningKey      OPTIONAL、
        archiveOptions      [1] PKIArchiveOptions   OPTIONAL、
        publicationInfo     [2] PKIPublicationInfo  OPTIONAL、
        oldCertId           [3] CertId              OPTIONAL
        -- id. of cert. which is being updated by this one }

POPOSigningKey ::= SEQUENCE {
        alg                     AlgorithmIdentifier、
        signature               BIT STRING
        -- the signature (using "alg") on the DER-encoded
        -- POPOSigningKeyInput structure given below }
POPOSigningKeyInput ::= SEQUENCE {
        authInfo                CHOICE {
        sender                  [0] GeneralName、
        -- from PKIHeader (used only if an authenticated identity
        -- has been established for the sender (e.g.、 a DN from a
        -- previously-issued and currently-valid certificate)
        publicKeyMAC            [1] BIT STRING
        -- used if no authenticated GeneralName currently exists for
        -- the sender; publicKeyMAC contains a password-based MAC
        -- (using the protectionAlg AlgId from PKIHeader) on the
        -- DER-encoded value of publicKey }
        publicKey               SubjectPublicKeyInfo    -- from CertTemplate            }

PKIArchiveOptions ::= CHOICE {
        encryptedPrivKey        [0] EncryptedValue、
        -- the actual value of the private key
        keyGenParameters        [1] KeyGenParameters、
        -- parameters which allow the private key to be re-generated
        archiveRemGenPrivKey [2] BOOLEAN
        -- set to TRUE if sender wishes receiver to archive the private
        -- key of a key pair which the receiver generates in response to
        -- this request; set to FALSE if no archival is desired. }

PKIPublicationInfo ::= SEQUENCE {
        action      INTEGER {
        dontPublish (0)、
        pleasePublish (1) }、
        pubInfos  SEQUENCE OF SinglePubInfo OPTIONAL
        -- pubInfos should not be present if action is "dontPublish"
        -- (if action is "pleasePublish" and pubInfos is omitted、
        -- "dontCare" is assumed) }
SinglePubInfo ::= SEQUENCE {
        pubMethod     INTEGER {
        dontCare    (0)、
        x500        (1)、
        web         (2) }、
        pubLocation   GeneralName OPTIONAL }
```

```
CertId ::= SEQUENCE
        issuer          GeneralName、
        serialNumber    INTEGER }

CrossCertRepContent ::= CertRepContent
CertRepContent ::= SEQUENCE
        caPub           [1] Certificate             OPTIONAL、
        response            SEQUENCE OF CertResponse }

CertResponse ::= SEQUENCE
        certReqId           INTEGER、
        -- to match this response with corresponding request
        status              PKIStatusInfo、
        certifiedKeyPair    CertifiedKeyPair    OPTIONAL}

PKIStatusInfo ::= SEQUENCE {
        status      PKIStatus、
        failInfo    PKIFailureInfo    OPTIONAL }
PKIStatus ::= INTEGER
        granted                 (0)、
        -- you got exactly what you asked for
        grantedWithMods         (1)、
        -- you got something like what you asked for; the
        -- requester is responsible for ascertaining the differences
        rejection               (2)、
        -- you don't get it、 more information elsewhere in the message
        waiting                 (3)、
        -- the request body part has not yet been processed、
        -- expect to hear more later
        revocationWarning       (4)、
        -- this message contains a warning that a revocation is
        -- imminent
        revocationNotification (5)、
        -- notification that a revocation has occurred
        keyUpdateWarning        (6)
        -- update already done for the oldCertId specified in
        -- FullCertTemplate }

PKIFailureInfo ::= BIT STRING
        -- since we can fail in more than one way!
        badAlg          (0)、
        badMessageCheck   (1)
        -- more TBS}

CertifiedKeyPair ::= SEQUENCE
        certificate     [0] Certificate         OPTIONAL、
        encryptedCert   [1] EncryptedValue      OPTIONAL、
        privateKey      [2] EncryptedValue      OPTIONAL、
        publicationInfo [3] PKIPublicationInfo    OPTIONAL}

EncryptedValue ::= SEQUENCE
        encValue            BIT STRING、
        -- the encrypted value itself
        intendedAlg     [0] AlgorithmIdentifier   OPTIONAL、
        -- the intended algorithm for which the value will be used
        symmAlg         [1] AlgorithmIdentifier   OPTIONAL、
        -- the symmetric algorithm used to encrypt the value
        encSymmKey      [2] BIT STRING              OPTIONAL、
        -- the (encrypted) symmetric key used to encrypt the value
```

```
        keyAlg          [3] AlgorithmIdentifier    OPTIONAL
        -- algorithm used to encrypt the symmetric key }

PKIConfirmContent ::= NULL

PKIProtection ::= BIT STRING
ProtectedPart ::= SEQUENCE
        header      PKIHeader、
        body        PKIBody}
```

3.個別技術編

3.1.ＳＥＴ相互認証技術

Secure Electronic Transaction（ＳＥＴ）は米国のビザとマスターカードとが共同で開発したカード決済プロトコルで、認証プロトコルと支払プロトコルの２つを定義している。両社はＳＥＴに対する意見を求める広く目的でインターネット上でそのドラフト版を公開している。

ＳＥＴの仕様は、平成９年３月時点でもまだ確定していない。また、ＳＥＴでは多くの仕様がオプションの扱いになっている。これらのオプションの実装が異なる製品間ではその相互接続も難しいと想定される。(例えばカード所有者が認証書を持つかどうかがオプションになっている。)

ＳＥＴの相互認証を検討する前に、これらのオプションの実装を事前に相互に確認される事を推奨する。

ＳＥＴの認証プロトコルでは各ブランドごとに閉じた階層型の認証局構成を前提としている。最上位のルート認証書を何らかの方法ですべてのエンドシステムに事前に組み込むことを前提とし、支払プロトコルにおいて相手の認証書を最初に検証する。相手認証書の検証は、認証局の認証書およびその上位のすべての認証書をルート認証書まで検証する。各カードブランドごとに閉じた認証システムを構成し、異なるカードブランド間の相互接続は対象としていない。このためＳＥＴには相互認証という考え方はなく、相互認証に関しては何も規定されていない。

この章ではまずＳＥＴで仮に相互認証を行うとした場合を想定した技術的な検討課題を整理する。次にこの検討課題を解決することができるとして、ＳＥＴにおける相互認証の構成及び相互認証書の形式を定義する。さらに、ＳＥＴにおける相互認証書の交換手順と支払プロトコルにおける相互認証書による認証手順について考察する。

3.1.1.ＳＥＴ相互認証の課題

カード決済の現実世界を考えると、消費者側のカード所有者（ＣＨ）はある特定のカードブランドに加盟している。単一または複数のブランドに加盟できるが、カード決済の時には、どのカードブランドのカードで決済するかをカード所有者が決定しなければならない。一方、商店側の販売店（Ｍ）は通常複数のカードブランドと契約している。販売店の規模にもよるが、理想的には主要なカードブランドのすべてに加盟する。

この現実世界でも、異なるカードブランド間の決済は想定されていないが、日本ではインターナショナル兼用のカードブランドの幾つかは、インターナショナルブランドが同じ場合には、日本のブランドが異なっている場合でも決済が可能である。

(1) ブランドの一致性に関する課題

ＳＥＴの支払プロトコルでは最初の初期手順でカードブランドＩＤをカード所有者（ＣＨ）が販売店（M）に対して通知する。販売店（M）ではこのカードブランドＩＤに加盟しているかどうかを検証する。もし販売店（M）がカード所有者のカードブランドに加盟していない場合には支払い取引き不成立になると考えられる。もし加盟している場合には認証経路は両者で一致するため相互認証は使用しない。

(2) 認証書の組織名

ブランドごとに閉じている現状の認証モデルを前提とするＳＥＴでは、ブランドと認証書とが対応している。ＳＥＴではエンドシステム認証書の所有者名における X.500 識別名（ＤＮ）の組織名（Ｏ＝）としてブランドＩＤを使用する。エンドシステム認証書の認証書発行者名（認証局）の組織名は認証局の組織名を使用する。認証書の検証においても相手組織名を事前に検証している可能性がある。

(3) ＳＥＴ認証局の構成

ＳＥＴの認証局はブランド全体を管轄するブランド認証局（ＢＣＡ）とその上位に位置するルート認証局（ＲＣＡ）から構成される。ブランドＣＡの下位に中間の認証局（オプション）である地域政策的ＣＡ（ＧＣＡ）があり、その下位にエンドシステムを認証する３種類のＣＡがある。カード所有者認証書を発行するＣＣＡ、販売店認証書を発行するＭＣＡ、アクワイアラ支払ゲートウェイ認証書を発行するＰＣＡから構成される。

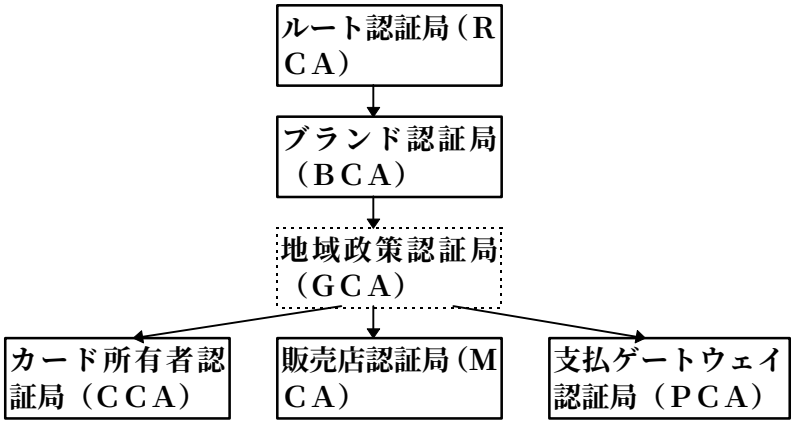


図２－１　ＳＥＴ認証局の構成

3.1.2. ＳＥＴ相互認証の構成

ＳＥＴ相互認証を行うとした場合の認証局システム構成について提案する。

ＳＥＴではブランドＩＤの一致がカード支払プロトコルの前提になっている。このため、異なるブランド間の相互認証を実現することが不可能と考えられる。ＳＥＴで相互認証が可能なのは、同一ブランド内での支払取引で最短経路による相互認証のケースに限定される。このシステム構成例はつぎの通りである。

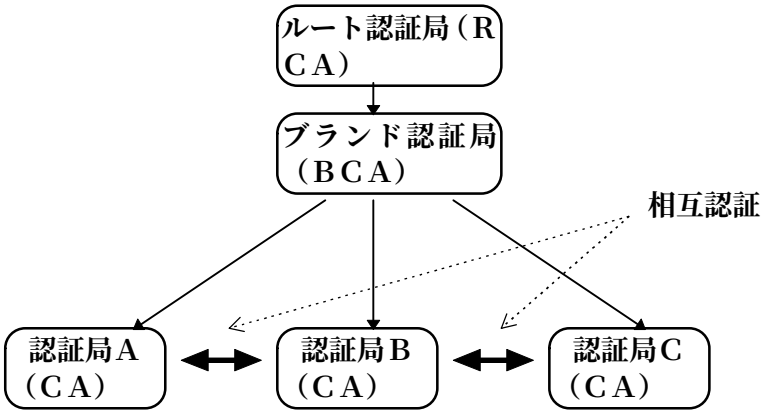


図２－２　ＳＥＴ相互認証システム構成例

3.1.3. ＳＥＴ相互認証書の形式

Ｘ．５０９　Ｖ３で定義される部分の相互認証書の形式は、共通技術編の１．４．３ 相互認証書の実装規約で規定している内容と同じである。ここではＳＥＴの仕様として独自に拡張された、ＳＥＴ個別拡張部の相互認証書での形式についてまとめる。

表２－１　認証書ＳＥＴ個別拡張部の構成

SET 個別拡張部	適用	クリティカル	拡張詳細情報	属性
ハッシュルート鍵 Hashed Root Key	ルート認証書	認識 (YES)	ダイジェストアルゴリズム	ＯＩＤ
			ルート鍵の指紋	OctetString
認証書のタイプ Certificate Type	すべての認証書	認識 (YES)	認証書の種類 (認証局は複数指定可能)	BIT STRING
販売店データ Merchant Data	販売店認証書	無視 (NO)	販売店識別子	CharacterString
			販売店アクワイアラＢＩＮ	CharacterString
			販売店の名前	CharacterString
			販売店の所在都市名	CharacterString
			販売店の所在都道府県名	CharacterString
			販売店郵便コード	CharacterString
			販売店所在国名	CharacterString
			販売店承認フラグ	BIT STRING
カード所有者認証要否 Cardholder Certificate Required	支払GW認証書	無視 (NO)	カード所有者認証フラグ (認証書を持たないカード所有者のサポート可否)	Boolean
トンネリング Tunneling	支払GW認証局	無視 (NO)	トンネリングサポート トンネリングアルゴリズム	Boolean ＯＩＤ
ＳＥＴの格付け SET Qualifier	ルートを除くすべての認証書	認識 (YES)	ポリシー宣言掲示ＵＲＬ	IA5 String
			ポリシー宣言 Email	IA5 String
			ハッシュアルゴリズム ポリシー	ＯＩＤ
			ポリシーダイジェスト	OctetString
			認証書発行ポリシー	IA5 String

(1) ハッシュルート鍵

ルート鍵のハッシュ（指紋）はルート認証書でのみ適用される。ＳＥＴ相互認証では異なるルート認証局間の相互認証は有り得ない。実装規約では対象外とする。

```
hashedRootKey EXTENSION ::= { -- Only in root certificates
        SYNTAX          HashedRootKeySyntax
        IDENTIFIED BY   { id-set-hashedRootKey } }

HashedRootKeySyntax ::= RootKeyThumb

RootKeyThumb ::= SEQUENCE
        digestAlgorithm     DAlgorithmIdentifier -- (sha1)--、
        rootKeyThumbprint   Digest }
```

(2) 認証書のタイプ

認証書のタイプはすべての認証書で適用される。相互認証書においてもこの情報を適用する。

```
certificateType EXTENSION ::=
        SYNTAX              CertificateTypeSyntax
        IDENTIFIED BY       { id-set-certificateType } }

CertificateTypeSyntax ::= BIT STRING
        card (0)、
        mer (1)、
        pgwy (2)、
        cca (3)、
        mca (4)、
        pca (5)、
        gca (6)、
        bca(7)、
        rca (8)、
        acq (9) }
```

(3) 販売店データ

この情報は販売店の認証書でのみ適用される。ＣＡの相互認証では対象外である。

```
merchantData EXTENSION ::=
        SYNTAX              MerchantDataSyntax
        IDENTIFIED BY       { id-set-merchantData }}

MerchantDataSyntax ::= SEQUENCE
        merID               MerchantID、
        merAcquirerBIN      BIN、
        merName             DirectoryString { 25 }、
        merCity             DirectoryString { 13 }、
        merStateProvince    DirectoryString { 3 }、
        merPostalCode       DirectoryString { 14 }、
        merCountry          DirectoryString { 3 }、
        merAuthFlag         BOOLEAN DEFAULT FALSE }
```

(4) カード所有者認証要否･トンネリング

これらの情報は支払ＧＷでみ適用される。 ＣＡの相互認証では対象外である。

```
cardCertRequired EXTENSION ::=
        SYNTAX              BOOLEAN
        IDENTIFIED BY       { id-set-cardCertRequired }}

tunneling EXTENSION ::=
        SYNTAX              TunnelingSyntax
        IDENTIFIED BY       { id-set-tunneling } }

TunnelingSyntax ::= SEQUENCE
        tunneling           BOOLEAN DEFAULT FALSE、
        tunnelAlgIDs        TunnelAlg }

TunnelAlg ::= SEQUENCE OF OBJECT IDENTIFIER
```

(5) ＳＥＴの格付け

認証局のポリシーに関連する付加情報である。相互認証書においてもこの情報を適用する。

```
setQualifier EXTENSION ::=
        SYNTAX              SETQualifierSyntax
        IDENTIFIED BY       { id-set-setQualifier } }

SETQualifierSyntax ::= SEQUENCE
        policyDigest        OCTET STRING (SIZE(16..20))、
        digestAlgorithm     DAlgorithmIdentifier、
        terseStatement      IA5String (SIZE(1..150))、
        policyURL  IA5String、
        policyEmail         IA5String }
```

表２－２　相互認証書ＳＥＴ個別拡張部の実装規約

SET 個別拡張部	適用	クリティカル	拡張詳細情報	相互認証実装規約
認証書のタイプ Certificate Type	すべての認証書	認識 (YES)	認証書の種類 (認証局は複数指定可能)	○
ＳＥＴの格付け SET Qualifier	ルートを除くすべての認証書	認識 (YES)	ポリシー宣言掲示ＵＲＬ	○
			ポリシー宣言 Email	○
			ハッシュアルゴリズムポリシー	○
			ポリシーダイジェスト	○
			認証書発行ポリシー	○

3.1.4. ＳＥＴ相互認証書交換手順

ＳＥＴではエンドシステムと認証局間の認証書発行手順を規定しているが､認証局間の認証書の発行手順については対象外としている｡相互認証における相互認証書の交換は認証局間で行われるため､以下に示す規定はＳＥＴの仕様には直接の影響をもたらすものではない｡しかし､認証局の実装を容易にするため､仮想的に一方の認証局がエンドシステムのプロトコル相当の処理を実行する方式を前提とする｡

ＳＥＴの認証プロトコルでは､非同期型（メール等）と同期型（WWW等）の２種類の手順が定義されている｡いずれの形態も適用可能である｡

(1) 非同期型相互認証プロトコル

双方の認証局が非同期に認証要求を発行し､それに対して双方が非同期に認証応答を返する｡事前準備に関しては本資料の対象外である｡双方ともセキュアなメールを使用することを推奨する｡

図２－３ 非同期型相互認証手順

(2) 同期型相互認証プロトコル

一方の認証局が相互認証手順を開始する｡エンドシステムの場合と同様に認証開始手順を起動してから認証要求を開始する｡

図２－４ 同期型相互認証手順

3.1.5.相互認証書による支払認証手順

カードホルダーと販売店、支払ゲートウェイ間で実行されるＳＥＴの支払手順において、相手認証書の確認を行う。ＳＥＴでは支払開始手順で要求側（カードホルダー）が販売店に対して自認証書のセット（認証局の認証書を含む）を渡す。販売店はこの認証書のセットを使って認証ルートを検証する。この検証結果が可であれば、 販売店は同様に自認証書のセット（認証局の認証書を含む）をカードホルダーに渡す。カードホルダーもこの認証書のセットを使って認証ルートを検証する。

この認証書のセットの中に、相互認証書を含む場合の認証ルートの検証方法についてまとめる。ＳＥＴでの認証ルート検証は通常ルート認証書まで確認するが､相互認証による検証は次の手順で行う。

【販売店が行うカードホルダーの検証】

① 相手認証書のセットの中から、最初にカードホルダーの認証書を検証する。

② 次にカードホルダー認証書を発行した認証局の発行者名と、販売店自身が保有する認証局相互認証書の発行者名とを比較する。一致する相互認証書を検索する。

③ ②で一致する相互認証書が存在する場合、次に認証局相互認証書の所有者名がカードホルダー認証書の発行者名と一致するものがあるかどうかを検索する。

以上の手順のすべてで可となった場合には、カードホルダーの認証は完了したとみなす。

この手順は共通編で述べた、最短経路相互認証手順に相当する。

【カードホルダーが行う販売店の検証】

① 相手認証書のセットの中から、最初に販売店の認証書を検証する。

② 次に販売店認証書を発行した認証局の発行者名と、 カードホルダー自身が保有する認証局相互認証書の発行者名とを比較する。一致する相互認証書を検索する。

③ ②で一致する相互認証書が存在する場合、次に認証局相互認証書の所有者名が販売店認証書の発行者名と一致するものがあるかどうかを検索する。

以上の手順のすべてで可となった場合には、 販売店の認証は完了したとみなす。

この手順は共通編で述べた、最短経路相互認証手順に相当する。

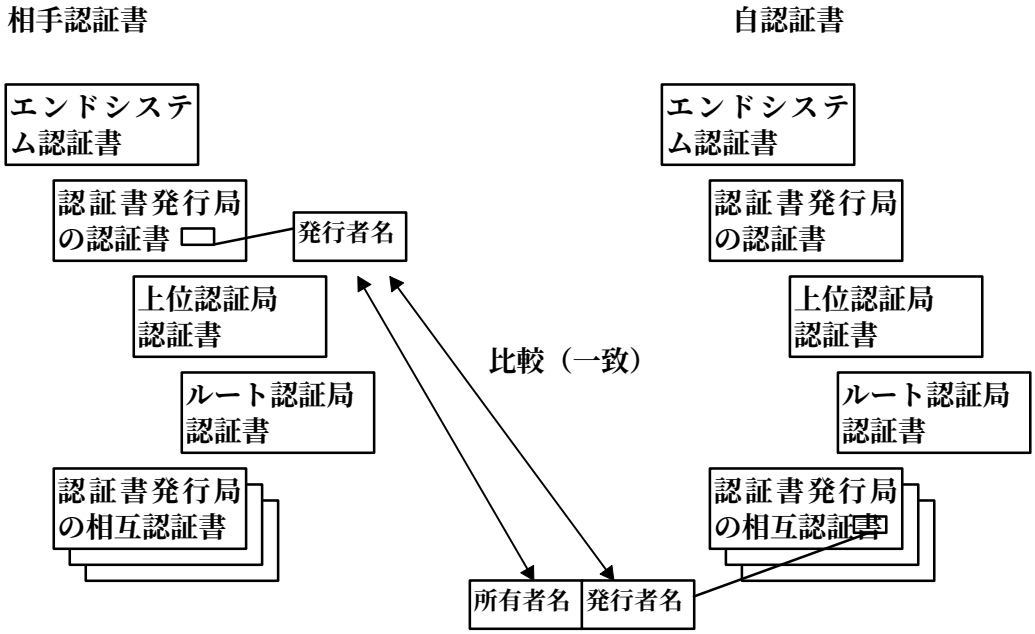


図２－５ 支払手順における相手認証書検証方法

3.2. ＳＳＬ相互認証技術

この章では Netscape Communications 社の開発した Secure Socket Layer(SSL) で使用する認証書における相互認証について考察している。ブラウザを利用するアプリケーションに依存しない共通のセキュリティプロトコルの一つとして開発されたＳＳＬはほとんどのWWWクライアント、サーバ製品に実装されている。ＳＳＬプロトコルはＩＥＴＦにドラフト仕様として公開されている。(SSL Version3)[18] しかし、ＳＳＬで使用する認証書の認証手順についての実装方法は対象外となっている。この資料では Netscape Communications 社の実装方法を前提とした相互認証について検討している。ブラウザ、サーバの製品開発元が異なる場合には実装方法が異なるかもしれない。この場合の相互認証は本資料の対象外である。

Netscape Communications 社の実装方法は同社の WWW サイトで公開されている。[19] 以下の技術解説はこの公開資料を参考にしている。

ＳＳＬのプロトコルには V2 と V3 の２つのバージョンがあるが、以下の説明はバージョン３を前提とする。 ＳＳＬプロトコルでは認証書をクライアント、サーバが使用することはオプションとなっているが、この資料ではクライアント、サーバの両方が認証書を使い互いに相手を認証することを前提とする。

[18] IETF Draft “The SSL Protocol Version 3.0” <draft-freier-ssl-version3-02.txt> (96/11/18)

[19] “Securing Communications on the Intranet and over the Internet” http://home.netscape.com/newsref/ref/128bit.html

3.2.1. ＳＳＬ認証技術

ＳＳＬプロトコルでは次の手順で認証書を交換する。

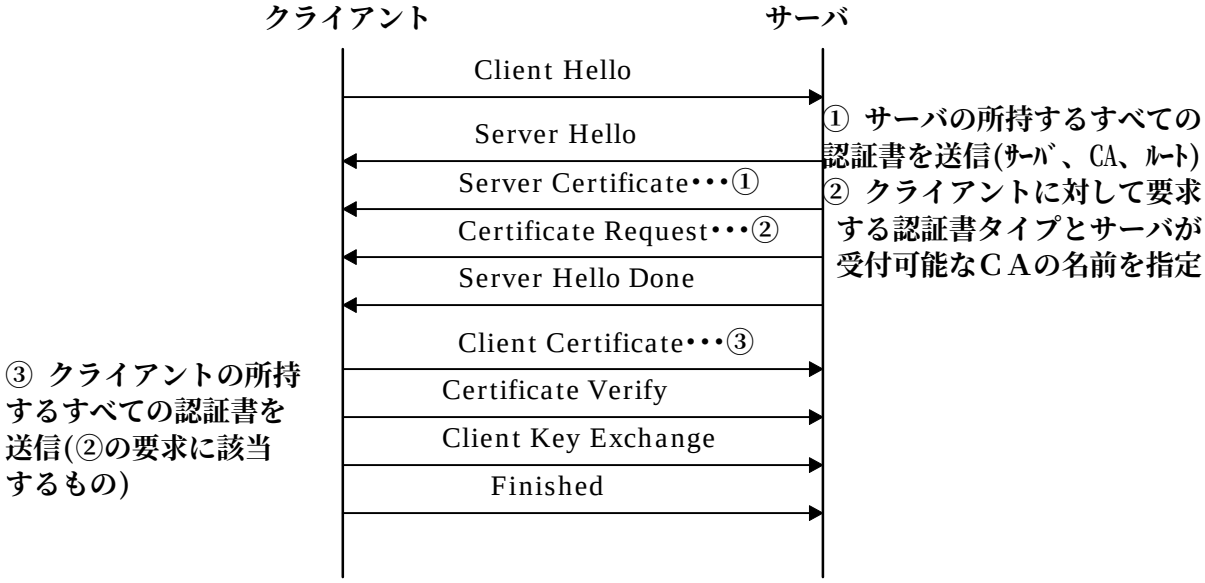


図 3.-6 ＳＳＬプロトコル認証書交換手順

(1) ＳＳＬプロトコルで交換する認証書は Certificate Chain と定義されているので、認証書の検証は階層型のルート認証書までの検証を行う方式と推定される。

(2) クライアント(ブラウザ)側がサーバの認証書を検証する手段は次の２つがある。

① ブラウザにあらかじめ幾つかのＣＡが登録されている。これらの情報は Netscape 社があらかじめソフトウェア出荷前に組み込んだ信頼できるＣＡの証明書である。しかし登録されていないＣＡの認証書をユーザ定義により受け付けることも可能である。

② さらにブラウザはサーバの認証書の所有者名がサーバのＵＲＬと一致する事を確認する。

(3) サーバ側がクライアントの認証書を検証する方法は、ＳＳＬプロトコルでサーバが受付可能なＣＡを指定することで、ある程度制限されると推定される。サーバ側がクライアント認証書を検証する方法として次の２つが紹介されている。

① クライアントの認証書をアクセス制御データベースのエントリィとして管理者がマップすることができる。

② クライアント認証書をＮＳＡＰＩ経由及びＣＧＩ環境変数としてサーバ側のアプリケーションに渡すことができる。

3.2.2. ＳＳＬ認証技術の課題

ＳＳＬの認証技術については共通仕様が定義されていないので、ＳＳＬの相互認証は製品ごとに実装レベルの検証が必要と考えられる。ここではＳＳＬ開発元の代表的な公開情報をベースに技術課題をまとめる。

(1) ブラウザに登録するサーバ認証ＣＡ

クライアントが検証できるサーバ認証書のＣＡ認証書があらかじめブラウザソフトに組み込まれている事を前提としている。認証局認証書が信頼できるサイトとしてブラウザに組み込まれるための審査基準等は明確になっていない。ブラウザソフトを開発した会社の信頼性がこの仕掛けの前提となっている。ＳＳＬの認証書を受け取るサーバは、サーバが前提とするブラウザにサーバ認証書を発行する認証局が登録されているかどうかを事前に検証する事を推奨する。

(2) ブラウザに登録されていないＣＡ

一時的あるいは何等かの理由でブラウザに登録されていないＣＡをクライアントユーザの判断で検証することができる。ただし、この機能は他のブラウザ製品ではサポートされていない。

この機能はユーザがＧＵＩを通してブラウザの指示に従って検証できるようになっている。

信頼できる別の手段によってＣＡが信頼できるサイトとして認められている場合には、この方法を使うことができる。(例えば新聞等で公的機関が公開するなど)

(3) サーバのクライアント認証ＣＡ

サーバはクライアント要求プロトコルによりクライアントに期待するＣＡ認証書を指定することができる。この要求に合うＣＡ認証書をクライアントが所持していない場合が考えられる。サーバが期待するＣＡ認証書を指定するのはＳＳＬ用のクライアント認証書をブラウザが自動選択できる様にする目的である。クライアント認証を行うＳＳＬサーバは前提とするブラウザに組み込まれる認証書のＣＡを事前に別の方法で知る必要がある。

3.2.3. ＳＳＬ相互認証の実装

ＳＳＬでの認証は認証のチェインを階層型にたどる方式と推定される。また、クライアント、サーバの双方が事前に信頼できるＳＳＬ認証局を登録することが標準になっている。ＳＳＬにおいても標準的にはＣＡの相互認証を前提としていない。ＳＳＬで相互認証を行うにはある程度ブラウザ及びサーバの実装を相互認証用に規定する必要がある。以下の考察で述べる内容は、ブラウザやサーバの実装の必要条件を記述するが、このような実装の可否については各製品ごとに検討する必要がある。

(1) クライアント(ブラウザ)によるサーバの認証

サーバ認証局がクライアントに登録されていない場合の相互認証の可能性について考察する。

【ケース１】クライアント認証書に相互認証書を付加する。

クライアントが認証書を受け取る時に、認証局が相互認証している認証局の相互認証書を付加する方式が考えられる。順方向と逆方向のいずれかの相互認証書があれば相互認証書によるサーバ認証が可能である。

・逆方向相互認証書はサーバ要求するＣＡの認証書として代用できる。

相互認証書でも代用できる様にブラウザを実装する。

・順方向相互認証書は認証書の発行者名がサーバの要求するＣＡであれば検証可能である。

サーバが要求するＣＡの認証書の中から所有者名だけでなく発行者名も検索する様にブラウザを実装する必要がある。

【ケース２】サーバがサーバ認証書に相互認証書を付加する。

サーバがサーバ認証書を受け取る時に、相互認証書を受け取る方式が考えられる。サーバはサーバ認証書プロトコルで相互認証書も含めてクライアントに転送する。

クライアントは相互認証書も含めてケース１と同様にサーバ認証をする。

(2) サーバによるクライアントの認証

サーバの要求する認証局の認証書をクライアントが持たない場合の相互認証の可能性について考察する。

【ケース１】クライアントが所持する相互認証書を利用する。

クライアントが所持する相互認証書をサーバの要求する認証書として利用する方式が考えられる。順方向と逆方向のいずれかの相互認証書があれば相互認証書による代用が可能である。

クライアントはサーバの要求する認証局の認証書として相互認証書も利用できるように実装

する。サーバはクライアントが応答した相互認証書を期待する認証局の認証書として代用できる様に実装する。

【ケース２】サーバが保持する相互認証書を利用する。

クライアントがサーバ認証書を受け取る時に、相互認証書も受け取る方式が考えられる。サーバはサーバ認証書プロトコルで相互認証書も含めてクライアントに転送する。

クライアントは相互認証書で認証している認証局の認証書も応答可能とする。

サーバは相互認証書で規定している認証局の認証書も受付可とする。

3.2.4. ＳＳＬ相互認証書の形式

ＳＳＬで使用する認証書の形式としてＸ.５０９ V1、V2、V3 の何れも可能としている。

また、Netscape Communications 社の製品では独自の Netscape Certificate Extentinos が規定されている。相互認証書においてもこの規定に従うものとする。

X.509 の標準の範囲の実装については例題が示されているのみで、実装条件は特に規定されていない。本資料の共通技術編の１．４を参考にして頂きたい。

ＳＳＬ認証書としての実装条件は本資料でも規定しない方針とする。

Netscape Certificate Extentinos として規定されている内容をここでは紹介する。

表 3.-3 Netscape Certificate Extentinos

パラメタ	内容	形式
Netscape 認証タイプ	クライアント、サーバ、ＣＡ区分	BITSTRING
Netscape ベースＵＲＬ	認証書で使用されるベースとなる名前	IA5STRING
Netscape 無効ＵＲＬ	ベースＵＲＬの次に来る無効ＵＲＬ	IA5STRING
Netscape ＣＡ無効ＵＲＬ	ＣＡ認証書のみで有効な無効ＵＲＬ	IA5STRING
Netscape 更新ＵＲＬ	ベースＵＲＬの次の更新されたＵＲＬ	IA5STRING
Netscape ＣＡポリシーＵＲＬ	ＣＡが認証書を発行するポリシーを掲載するＵＲＬ	IA5STRING
Netscape ＳＳＬサーバ名	クライアントがサーバ認証で検証するサーバ名	IA5STRING
Netscape コメント	ユーザに認証書を表示する時のコメント	IA5STRING

4.付録Ａ．参考資料

1. ITU X.501:1988 THE DIRECTORY - THE MODELS
2. ITU X.509:1988 THE DIRECTORY - AUTHENTICATION FRAMEWORK
3. ISO/IEC 9594-8:1993 (E)、 The Directory: Authentication Framework
4. Amendment 1 to ISO/IEC 9594-8:1995 (E) : Certificate Extentions
5. Amendment 1 to ISO/IEC 9594-7:1995 (E) : Certificate Extentions
6. Amendment 2 to ISO/IEC 9594-6:1995 (E) : Certificate Extentions
7. Amendment 4 to ISO/IEC 9594-2:1995 (E) : Certificate Extentions
8. IETF Draft : Intenet Public Key Infrastructure Part1: X.509 Certificate and CRL Profile(96/12)
9. IETF Draft : Internet Public Key Infrastructure Part3: Certificate Management Protocols (96/12)
10. NIST Draft : Minimum Interoperability Specifications for PKI Components (MISPC) (96/12)
11. Federal Public Key Infrastructure(PKI) Technical Specifications
12. Part A - Requirements(96/1)
13. Part B - Technical security Policy(96/1)
14. Part C - Concept of Operations(95/11)
15. Part D - Interoperability Profiles(95/9)
16. Secure Electronic Transaction (SET) Specification Book2: Programmer's Guide(96/6)
17. IETF Draft : The SSL Protocol Version 3.0 (96/11)

5.付録Ｂ．検討メンバーリスト

ＥＣＯＭ

米倉 昭利　　主査　　電子商取引実証推進協議会 主席研究員

長　 博連　　副主査　 電子商取引実証推進協議会 主席研究員

角間 和博　　副主査　 電子商取引実証推進協議会 主席研究員

有識者

菊池 浩明　　東海大学 講師

リーダー・サブリーダー

西村 和郎　 リーダー　　株式会社日立製作所 ソフトウェア開発本部計画部 主任技師

鍛冶 俊彦　 サブリーダー　富士通株式会社 ニュービジネス推進部 プロジェクト課長

メンバー

池田 伸次　　株式会社シー・アイ・シー 電子取引研究プロジェクトチーム 主任研究員

越湖 正道　　株式会社日本ダイナースクラブ 企画部 上席調査役

大谷 彰宏　　三菱電機株式会社 Ｃ／Ｓネットワークシステム部

加藤 佳実　　株式会社ジャストシステム 技術本部開発

北井 富士夫　株式会社東芝 電算機ソフトウェア部

鈴木 雅人　　日本ベリサイン株式会社 エンジニアリンググループ エンジニア

竹永 三郎　　財団法人関西情報センター 主席研究員

松谷 英夫　　財団法人情報処理相互運用協会 技術部 第２技術課長

平成９年３月発行

発行：電子商取引実証推進協議会

東京都江東区青海２－４５
タイム２４ビル１０階
Tel　03-5531-0061
E-mail　info@ecom.or.jp