SCIS'98 The 1998 Symposium on
Cryptography and Information Security
Hamanako , Japan , January 28-31 , 1998
The Institute of Electronics ,
Information and Communication Engineers

WIDE プロジェクトにおける CA 運用実験と考察

Management and Study of experimental CA in WIDE Project

櫻井 三子[†] 服部 裕之[‡]

Mine Sakurai Hiroyuki Hattori

[†]日本電気株式会社 ネットワーキング技術研究所 [‡]明治大学 情報科学センター

〒 108 東京都港区芝浦 2-11-5

[†]NEC Corporation [‡]Meiji University

2-11-5 Shibaura , Minato-ku , Tokyo , 108 JAPAN

あらまし

公開鍵証明書を発行する際の手続きに着目し，既存のツールを使って WIDE プロジェクト内 CA の運用実験を行なった．実験では，WIDE プロジェクトメンバ限定の Web 情報に対して証明書を用いたアクセスコントロールを行なう前提で，事前の証明書発行手続きにおいて組織構造に沿ったメンバの本人確認をおこなった．これにより組織内 CA 運用のヒントを得ることができた．

abstract

This paper shows our experiment about a CA in WIDE Project , with existing tools , emphasized on an issuing procedure for public-key certificates . It is assumed that members of WIDE Project would need certificates to access to the members only Web pages . In the procedure , membership confirmation process was executed based on the organizational constitution , and this procedure could make some hints on managing internal CAs.

1. はじめに

CA (Certification Authority) 技術は通信相手が名乗ったときにそれが正しいかどうかを確認するための技術 (認証技術) の 1 つで，公開鍵暗号の公開鍵とユーザ識別子の対応を第 3 者が証明することで相手の正当性を保証する技術である．一度証明された公開鍵は証明書として安全かつ広範囲に配布することが可能となるため，CA 技術は認証技術の中でも広域分散環境に適した技術として注目を浴びている．最近，インターネット上のユーザを対象とした電子決済への応用や S/MIME[3] などの電子メールのセキュリティ強化への応用，SSL[4](Secure Sockets Layer), TLS[5](Transport Layer Security) を用いた WWW のセキュリティ強化への応用が見られる．特に，Netscape 社製や Microsoft 社製など Web ブラウザのように一般的に広く受け入れられているツールが標準で SSL 対応になってきており，かなり多くのユーザが CA 技術を利用できる環境になりつつある．

しかし，CA 技術が既に一般に普及しているかというと，まだそのような段階にはない．現状はほとんどのシステムが実験段階である．しかも，CA やアプリケーションについての機能確認やパフォーマンス評価を目的とした実験が主体であり，CA の運用面に焦点をあてた実験はほとんど行なわれていない．CA 運用に関する各種ツールの開発が進んでも，CA が発行する証明書内のユーザ識別子を具体的に何にするか，ユーザ識別子と公開鍵の対応を証明するのにどのような情報を根拠とすべきか，といった点については CA を運用する各組織が証明対象に応じて検討しなければならないことであり，運用に関するケーススタディが強く求められている．

われわれは，インタネットの研究プロジェクトである WIDE プロジェクト (代表: 慶應義塾大学 村井純教授) メンバとして，研究活動を行なっている．WIDE プロジェクトでは，メンバ間が情報共有するための手段の 1 つとして Web サーバを運用しているが，メンバ限定の情報に対する保護方法はパスワード認証のみであり，さらなるセキュリティ強化について検討すべき時期にきている．例えば，SSL 対応 Web ブラウザや Web サーバを利用し，SSL のクライアント認証機能を用いることはセキュリティ強化方法の 1 つとして提案できる．SSL 対応 Web ブラウザが多くのユーザに利用できることを考えると，WIDE プロジェクトに参加する全メンバを対象にすることが可能であり，WIDE プロジェクトという組織全体のスケールを考慮した実験が行なえる．また，既存のツールを利用することで新しいアプリケーションの開発を必要とすることなく，CA 運用面に重点を置いた実験が可能である．

そこで，WIDE プロジェクトの参加メンバ限定の Web 情報へのアクセスコントロールを SSL のクライアント認証を用いて行なうために，セキュリティレベルと組織構造を考慮した CA の運用実験を行なった．

本稿では，まず証明書発行手続きを決定する上で考慮しなければならない WIDE プロジェクトの組織としての特徴についてまとめ，実験の前提となる Web のアクセスコントロールについて簡単に述べる．続いて，実験概要，結果および考察について述べる．

2. WIDE プロジェクトの組織的特徴

インタネットの研究プロジェクトである WIDE プロジェクトでは実験基盤となる WIDE インタネットの実運用，および，さまざまな分野と期間にわたる運用実験を行なっている．

2.1 構成員

WIDE プロジェクトは，おもに WIDE インタネットに接続している様々な大学や企業からの，学生や技術者から構成され，全体では 400 名を超える．WIDE プロジェクトのメンバとしての ID は，WIDE プロジェクト内のメーリングリストに登録される際の電子メールアドレスである．メンバには体制的に以下のような分類がある．

ボード 研究活動の他，WIDE プロジェクト全体の運営を行なう役割を持つメンバ．約 20 名．

一般メンバ ボードの会議で WIDE プロジェクトへの参加が了承されたメンバ．一般メンバには，入会，所属変更や脱会など

の際に連絡役となるボード(担当ボード)が必ず一人つく．

2.2 研究活動の形態

WIDE プロジェクトメンバは，おもに技術分野ごとのグループ(ワーキンググループ，WG) に分かれて活動している．ワーキンググループによっては時々オフラインミーティングを開いているが，WIDE プロジェクト全体で顔を合わせる機会はおおよそ以下の場合である．

- 1 日構成の定例研究会 (5,7,11,12 月)
- 合宿形式の研究会 (3,9 月)

ミーティングや研究会以外では，ワーキンググループ単位や，全 WIDE メンバのメーリングリストを運用して情報交換や議論を行なっている．また，Web サーバを運用して，研究会の詳細プログラム等の情報をメンバ間で共有している．

3. Web のアクセスコントロール

3.1 アクセスコントロールのレベル

Web サーバへのアクセスコントロールを実現するには，以下の 2 つのレベルが考えられる．

- クライアントコンピュータ単位でのアクセスコントロール
- ユーザ単位でのアクセスコントロール

ユーザ単位でのアクセスコントロールにおいて，ユーザが本物かどうかを識別する情報として，以下の 2 つがある．

- パスワード
- 証明書

証明書を利用したアクセスコントロールは，たとえば Web サーバおよび Web ブラウザ間の通信を SSL を用いることで可能となる．

3.2 SSL のクライアント認証機能を用いたアクセスコントロール

SSL は，Socket レベルでのデータの暗号化機能や，サーバ認証機能およびクライアント認証機能を提供するプロトコルである．

SSL は，実際のデータの暗号化に先立って，公開鍵暗号を用いてセッション鍵をサーバ・クライアント間で共有する．サーバおよびクライアントそれぞれの公開鍵情報は，X.509 証明書フォーマット [1][2] で交換する．X.509 証明書は，図 1 に示すように，単に公開鍵ばかりでなく，その所有者，発行者，有効期限などの情報も含む．また，発行者 (CA) によって内容が保証され，CA の秘密鍵による署名が付いている．クライアントとサーバはあらかじめ信頼する CA を決めておくという前提の元で，受け取った証明書がその CA によって署名されているかによって検証を行なう．

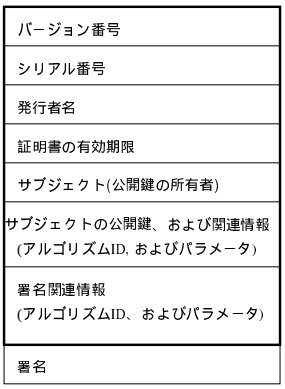


図 1: 証明書に含まれるデータ

SSL プロトコル上で Web のデータをやりとりする際には，Web サーバとブラウザに対して CA が証明書を発行し，Web サーバ側でアクセスを許可する証明書の情報をアクセスコントロールリストに登録することで，証明書を用いたアクセスコントロールが実現できる．例えば，SSLeay をベースとした SSL 対応 Web サーバでは，アクセスコントロールリストに図 2のように記述する．

4. 実験目的

WIDE プロジェクト内の CA を対象にした運用実験では，あらかじめ各メンバに証明書を発行し，メンバ限定の Web 情報提供サービスの中で証明書を利用するという前提をおく．この実験の目的は以下の 2 点であるが，特に前者に重点を置いた．

- 証明書発行において，メンバ限定の Web 情報提供サービスに求められるセキュリ

（例）/usr/local/etc/httpd/conf/.htpasswd ファイル

```
/C=JP/O=WIDE Project/CN=abc@def.meiji.ac.jp:xxj31ZMTZzkVA
```

これは，クライアントから提示された証明書の所有者名（サブジェクト）が，
/C=JP/O=WIDE Project/CN=abc@def.meiji.ac.jp
であるならばアクセスを許可するという意味である．

図 2: アクセスコントロールリストの例

ティレベルと，WIDE プロジェクトという組織の構造とを考慮した最適な手続きを提案し，実証すること

- 既存のツール（ブラウザ，Web サーバ，CA 運用パッケージ）を組み合わせて実験し，その機能限界と問題点を明確化すること

前者で，最適な手続きとはアプリケーションに必要なレベルの本人確認が達成でき，かつ，一連の作業がユーザに受け入れられる手続きである．前者に重点をおくことによって，CA 運用ツールだけでは実現困難なノウハウを収集し，他の組織で同様のサービスを行なう場合にも参考になるような一般的なノウハウを得ることを目標とする．

5. 実験概要

メンバ限定の Web 情報として 9 月の合宿研究会関連の情報をターゲットにした関係で，実験期間は 7 月の研究会直後から 10 月末までの 3ヵ月間とした．そして，実験期間の証明書発行ユーザ数の目標を合宿参加者数（= 約 230 名）とした．

本節では，実験に対する考え方，実験システムの具体的な構成，証明書発行手続きの詳細について述べる．

5.1 考え方

証明書を用いた合宿情報へのアクセスコントロールを実現するためには，アクセスを許可する者（= WIDE プロジェクトメンバ）に対し，事前に証明書を発行する必要がある．証明書発行時に最も重要となるのは，証明書を発行してよいユーザの本人確認手続きである．

今回のシステムではメンバ共通の情報を扱うため，メンバであることを確認できる程度の本人確認を行えばよい．その本人確認レベルは，組織でのメンバ登録時の本人確認レベルと同一となるべきである．メンバ登録時にメンバ証が発行されている場合はそれを確認するのがもっとも容易である．ない場合はメンバ登録手続きの再現を必要とする．再現の容易さは，本人確認に必要な情報の準備の容易さとその情報を確認する関係者の揃いやすさに影響を受けると予想される．WIDE プロジェクトでは，1) メンバ証はない，2) メンバ登録時の本人確認では特定の書類を必要としないが，メンバと担当ボードが顔を合わせるのが原則である．したがって，本人確認レベルは厳密ではないが，メンバの分散性の高さから，確認作業は困難なレベルにあると予想され，いくつかの手続きを提案し試すことにする．

5.2 システム構成

実験で利用した既存のツールは以下の通りで，これらを組み合わせて図 3に示す構成をとった．

- CA 運用サーバ
  ICAP1.0 [6]+ Web サーバ
- SSL 対応 Web ブラウザ
  Netscape Navigator 3.0 以上または Netscape Communicator 4.0 以上
- SSL 対応 Web サーバ
  Apache 1.2.+ SSLeay 0.6.6 ベースの SSL 対応パッチ

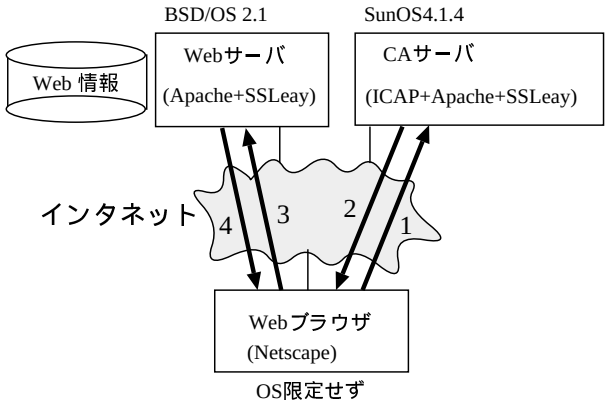


図 3: 実験のシステム構成

5.3 実験のステップ

一連の実験は2つのステップからなる．

ステップ1 ICAP を利用して CA サーバを運用し，X.509 形式の証明書を発行する (図3 の1,2)

ステップ2 Web サーバの SSL クライアント認証機能を用いて，証明書つきの SSL 対応 Web ブラウザから，アクセスコントロールをかけた Web ページにアクセスする (図3 の3,4)

各ステップにおいてメンバが行なった手続きを図4，5 にまとめる．

ステップ1 において，一般の発行手続きでは，本人確認後にその場で証明書を発行し配布する場合が多い．しかし，以下のようなツールの制約や特徴を考慮して，本人確認と同時に実験用アカウントとパスワードを配布し，後日そのアカウントを使って各自が証明書を発行する手続きにした．

- Web ブラウザ用の証明書を発行するためにはブラウザ側で鍵を作成する必要がある[1]が，本人確認するその場でブラウザを持っているとは限らない
- CA サーバ (ICAP) にはアカウント登録機能があり，登録されたユーザがパスワードを使って各自で証明書を発行することが可能である

[1]本人確認後に秘密鍵と証明書の対を渡したとしても，Netscape ブラウザにはそれを取り込む機能がまだなかった．

したがって，メンバの本人確認を行なった後に，安全な方法で 担当ボードから CA の運用者 (CA オペレータ) へ証明書発行用アカウントとパスワードを伝えて登録することにした (図6)．

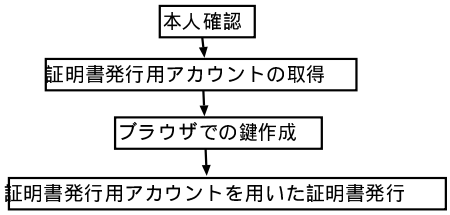


図 4: ステップ1 の流れ

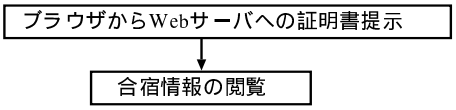


図 5: ステップ2 の流れ

5.4 本人確認手続き

実験にあたって，1) 誰が，2) いつ，3) どのような方法で「WIDE メンバである」ことを確認するかを決める必要があった．

1) については，節 2.1 を考慮し，各メンバの担当ボードが最適であると判断した．2) については，節 2.2で述べた定例研究会を利用して本人確認を行なうのが最も効率のよい機会である．しかし，全員が参加するわけではないため，メンバと担当ボードが随時本人確認を行なえるように決めた．3) については，WIDE プロジェクトのメンバ登録時に確認すべき書類が特に決められていないため，原則写真つき身分証明書類で確認することにしたが，詳細は各担当ボードに任せた．

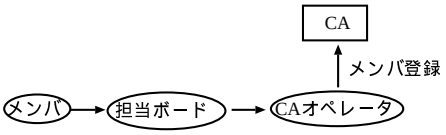


図 6: 本人確認の流れ

実際に行なった本人確認手続きは以下の3通りである．

(1) 7月の定例研究会にて
パスワードの書かれた用紙に氏名，電子メールアドレスなどを記入し，担当ボードがメンバの身元と用紙の記入事項を確認した後，CAオペレータの目の前で用紙に署名してからCAオペレータに手渡す．

(2) 9月合宿前日までの随時
メンバが担当ボードに電子メールアドレスとパスワードを何らかの方法で渡し，担当ボードがメンバを何らかの方法で確認し，担当ボードからCAオペレータにはPGPを利用してデータを暗号化し，オンラインで情報を渡す[2]．

(3) 9月合宿研究会にて
合宿会場受け付けや会議室にいるメンバがパスワードの書かれた用紙に氏名，電子メールアドレスを記入し，CAオペレータが身元と用紙の記入事項を確認する．

(1)はメンバ登録時の手続きに最も近い手続き，(2)はCAオペレータを含むメンバの分散性を考慮した手続き，(3)は手続きの行なわれる場所の物理的なセキュリティレベルを加味して本人確認を簡単にした手続きである．

5.5 証明書発行手続き

本人確認手続きの違いによらず，以下に示す手順で証明書を発行した．

i) 証明書発行アカウント登録通知を受けたメンバが，後日SSL対応Webブラウザを利用してCAサーバにアクセスする
ii) ガイドにしたがってCAサーバの証明書発行メニューを実行し，必要事項の記入や鍵作成を行なう
iii) 証明書発行アカウントとパスワードを用いて各自で証明書を発行し，ブラウザに証明書を格納する

[2](2)を実施する前には，あらかじめCAオペレータがボードの身元確認を行ない，各ボードのPGPの公開鍵が正しいことを確認した．

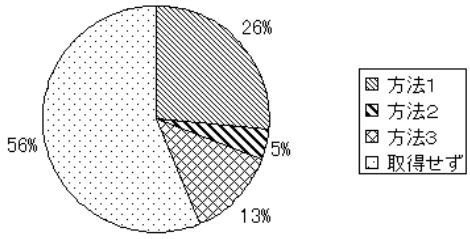
6. 実験結果

6.1 本人確認手続き

方法別の集計を表1に示す．合計数は180であり，目標数の230には満たなかった．

節5.4の方法(1)は多かったが，担当ボード本人が立ち会うことについてはうまくいかず，担当ボード以外のボードが代理で署名することが多かった．(2)はメンバからボードにパスワード情報を渡す方法を決めなかったことから，逆にメンバを戸惑わせることになり，予想以上に数が少ない結果となった．(3)は担当ボードを通さないため最も簡単な手続きであり，(2)よりも多かった．

表1: WIDEメンバ数(411)に占める証明書発行用アカウント取得の割合



6.2 証明書発行手続き

実験の結果，実際に証明書を発行したメンバ数を表2に示す．本人確認手続きを済ませたメンバのうち，証明書発行手続きまで至ったメンバは約7割の127名となった．

証明書発行が最も集中した際には1日で56件の証明書発行処理が行なわれたがCAサーバでの性能的な支障は特になかった．

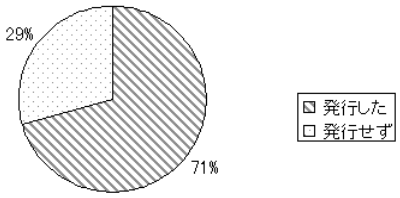
7. 考察

7.1 手続きに関する考察

7.1.1 本人確認手続き

節6.1で述べたように，「WIDEメンバであること」を確認するだけの比較的簡単な確認であったにもかかわらず，当初に決めた通りには実行できなかった．特に，担当ボードでない他のボードに本人確認の代行をお願いするケースが多く見られたが，ボードがメンバのことを知らない場合には身元確認ができ

表 2: 証明書発行用アカウント総数 (180) に占める証明書発行者数の割合



ても WIDE メンバかどうかを判定することは難しい．

WIDE のメンバ証があれば，メンバと CA オペレータの間のみでも確認手続きが行なえたはずである．しかし，メンバ証があったとしても，担当ボードか CA オペレータのどちらかがメンバと顔を合わせていないと確認は行なえない．メンバが物理的にも，所属組織的にも分散している環境では，本人確認手続きを短期間で行なうのは予想通り難しかった．

結果的には，組織構造が柔軟である場合，本人確認手続きの実行が困難であることが示唆された．しかし，今後新規に参加するメンバの本人確認手続きについては，メンバ登録手続きと連動させれば比較的容易になると考えられる．

7.1.2 証明書発行手続き

節 6.2 より，本人確認手続きを行なったメンバ数とその後証明書発行手続きまで行なったメンバ数にはかなりの差が見られる．実験期間中のメンバからの問い合わせ内容や，メンバに対して行なったアンケート調査から，証明書を発行する手順を示すガイドがわかりにくいという指摘が目立った．

特に，ブラウザの秘密鍵を保護するためのパスワードと ICAP に登録した証明書発行用アカウントのパスワードの使い分けが最も混乱を招いた点である．ガイドや ICAP インターフェイスを改善すれば使い分けることが可能なのか，証明書発行用アカウントを使って各自で証明書を発行するという方法自体を見直した方がよいのか，切り分けて検討する必要がある．

7.2 既存のツールの機能に関する考察

7.2.1 ICAP

ICAP は全てのソースを公開しているわけではないが，ユーザインターフェイスなどを変更することは容易である．[3]

結果として，100 人規模の CA を運用することができ，一日あたり最大 56 件の証明書発行処理が行なえることが確認できた．定常運用を考えると以下のような改善が必要である．

- CA の秘密鍵のセキュリティ管理
- 設定機能の充実化
  証明書の有効期限など組織の事情に合わせて変更すべきパラメータを簡単に変えられるようにするなど

7.2.2 SSL 対応 Web ブラウザ

Windows，UNIX を問わず様々な OS での動作が確認されたが，以下のような不安定な面がみられた．

- OS とブラウザの日本語モードの組み合わせによっては鍵の作成時にブラウザが無反応になる
- 複数の証明書を持っている場合に，複数の証明書から自動的に提示する証明書を選択する機能があったが，正しく動作せず強制終了する

7.2.3 SSL 対応 Web サーバ

機能的に不十分な点として，証明書の廃棄リスト (CRL) を処理する機能がない点が挙げられる．証明書は有効期間内に廃棄されることがあり，CA が定期的に CRL として発行することになっているが，Web サーバ側に CRL を処理する機能がないと，無効となるはずのクライアント証明書を有効と判定することになる．この問題は，CRL のチェックを行なうように，プログラムの改良を行なうことによって解決した．ただし，今回は試作のみで運用までには至らなかった．

[3] われわれは ICAP の開発にも携わっており，ユーザインターフェイス等の問題のいくつかは ICAP 次版で改良した．

また，今回 Web サーバの一部の情報に「WIDE の CA が発行した証明書を持つユーザは全てアクセス可能」というルールでアクセスコントロールをしようとした際に，以下のような不十分な点があった．

i) アクセスコントロールのポリシーが，Web ページ毎に設定できない
ii) アクセスコントロールリストに証明書の所有者名 (サブジェクト) しか記述できず，複数の CA を信用している場合にどの CA が発行した証明書なのかを厳密に指定できない
iii) アクセスコントロールリストに正規表現が使えず，アクセスを許可する人数分の DN 情報を記載しなければならない

i) については，アクセスコントロールをかけたい情報のために専用の Web サーバを運用して回避した．ii) については，Web サーバが信用する CA を 1 つに限定して問題を回避した．iii) については，Web サーバ管理者にとっては負担となることが予想されたため，図 2の中で正規表現を利用できるよう，プログラムの改良を行なって解決した．

8. おわりに

今回は，既存の CA 運用パッケージやアプリケーションを利用して現実にどの程度の運用やサービスができるかを試した．特に一般的なパッケージソフトでは実現困難な証明書発行の手続き面の実験を重視した．

証明書発行手続きが最適だったかどうかについては，評価分析中であり，明確な結論はまだ出ていない．但し，組織の柔軟性と本人確認手続きの困難さとの相関など有意な知見が得られている．実際，「WIDE メンバである」ことを確認するだけの，それほど厳密とはいえないレベルの本人確認であっても，当初に決めた通りには実行しきれなかった．本人確認後の証明書発行ではメンバに混乱が見られたが，問題点解消には，ツールの改良によるか，ガイドの強化によるか，あるいは CA 運用方法によって対処すべきかを，今後の実験で切り分けていく必要がある．

CA およびアプリケーションを実現する既存のツールの実用性については，動作不安定な点，機能的に不十分な点およびユーザインターフェイスの問題点がいくつかあり，一般ユーザ向けのサービスとして提供できるレベルではない．しかし，今回はブラウザを Netscape 社製に限定したにもかかわらず，100 名以上 (1/4 強) の実験参加を得ることができた．したがって，多数の参加者を見込んだ実験を行なうことが可能な環境になってきたといえる．

今後も手続き面を重視した実験を行ない，今回満足できなかった機能の追求や，今回は実験できなかった証明書の再発行手続きや，廃棄手続き等を試す予定である．

謝辞

本稿は，WIDE プロジェクトの moCA (members only CA) WG における実験をもとに作成しました．まず，実際に実験に参加して下さった WIDE プロジェクトメンバに深謝いたします．また，CA 運用パッケージ (ICAP) を利用した実験開始にあたり，様々なアドバイスをいただいた認証実用化実験協議会 (ICAT) の広域認証技術タスクフォースメンバに感謝いたします．さらに，本稿をまとめるにあたり，ご指導いただきました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のグループメンバに感謝いたします．

参考文献

[1] ISO/IEC JTC 1/SC 21: “Information Technology - Open Systems Interconnection , Data Management and Open Distributed Processing” , Final Text of the 1993 Edition of ISO/IEC 9594-8—ITU-T Rec X.509 (Jun . 1994)
[2] ISO/IEC JTC 1/SC 21/WG 4 and ITU-T Q15/7: “Information Technology - Open Systems Interconnection – The Directory: Authentication Framework , AMENDMENT 1: Certificate Extensions”, Proposed Draft Amendment 1 to ITU X.509 (Jul . 1995).
[3] http://www.rsa.com/rsa/S-MIME
[4] http://home.netscape.com/eng/ssl3/ draft302.txt
[5] Internet-Draft: “The TLS Protocol Version 1.0” , draft-ietf-tls-protocol-05.txt , (working draft) (Nov . 1997)
[6] 服部 裕之，櫻井 三子，小林 良至，菊池 浩明: “オンライン証明書発行局パッケージ (ICAP) の実装と評価”, SCIS97 (Jan . 1997).