マルチメディア通信と分散処理　106－7
コンピュータセキュリティ　16－7
（２００２．２．１４）

ＰＫＩにおけるDN照合規則のあり方に関する考察

今枝直彦　政本廣志　永吉剛
ＮＴＴ情報流通プラットフォーム研究所

ネットワーク上での認証の実現方法の１つとして、PKI を用いた認証がある。PKI においては公開鍵証明書の所有者を識別するため、CA への公開鍵登録やディレクトリエントリへの公開鍵証明書格納・参照，認証を必要とするアプリケーションにおいて名前の照合が必要になる。しかし、PKI で規定されている照合規則および上記照合場面における照合方法を現実社会に適用すると、“ディレクトリから目的の公開鍵証明書を取得できない”“アプリケーション所有データベースとの照合ができない”といった課題が発生する。そこで、本稿では、これらの課題を解決するために、公開鍵証明書に記載された同一の利用者を示すすべての名前のエントリに公開鍵証明書を格納する。また、ディレクトリの特定属性型にこれらの照合規則（新規照合規則も定義可能）を格納し、この照合規則を利用することで照合を行う方式について述べる。

A method to deal with Name Comparison in PKI

Naohiko IMAEDA　Hiroshi MASAMOTO Takeshi NAGAYOSHI
NTT Information Sharing Platform Laboratories

This paper proposes a method to solve tasks where name comparison rules and methods apply to PKI. To store certificates in all directory entry to show the same user, is used to get a certificate of the purpose by optional informations about verification target. To store a name comparison rule in the directory attribute type, which is possible to define a new name comparison rule, is used to grasp the Name comparison rule for a user which to be using with CA.

1. PKI における照合場面

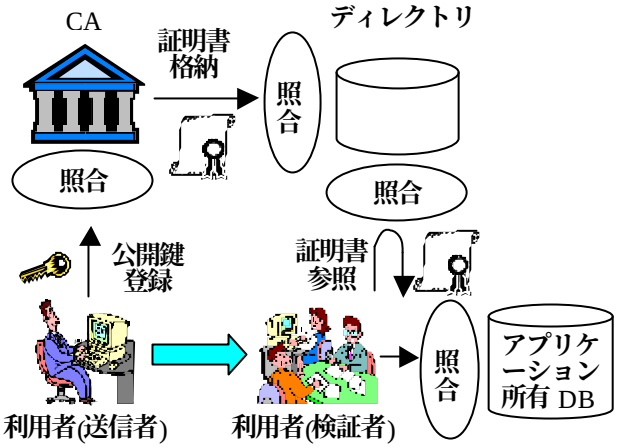

図１　PKI における照合場面

PKI を用いた認証を行うためには、以下の３つのプロダクトの４場面において名前の照合が必要になる。

１）CA への公開鍵登録
２）ディレクトリへの公開鍵証明書格納・参照
３）アプリケーションにおける照合

PKI を用いた認証を行う利用者は、CA に公開鍵登録申請を行うことにより CA から公開鍵証明書を取得する。この公開鍵登録時に、CA は申請された利用者の名前を示す PrintableString，UTF8String などさまざまな StringType を持つ DistinguishedName（DN）が既に登録されてい

るDNであるかを照合することで人や団体などの登場人物（Party）を一意に特定する。また、発行した公開鍵証明書をディレクトリ内の公開鍵証明書の格納場所を示すエントリに格納する(通常、公開鍵証明書のSubjectのDNエントリに格納)が、この公開鍵証明書のディレクトリエントリへの格納においてもCAが指定したDNとディレクトリ内に格納されている公開鍵証明書のエントリとの照合が必要になる。そして、CAから上記DNを公開鍵証明書のSubject（所有者名）としてもつ公開鍵証明書を受け取った利用者(送信者)は、実際の通信相手への文書送信時、文書とともにディジタル署名を検証者に送信する。このディジタル署名付き文書を受け取った検証者は、ディジタル署名を検証するために、上記ディレクトリエントリへアクセスする（通常、DNを指定）ことにより送信者の公開鍵証明書を参照した後、ディジタル署名の検証を行うとともに検証者の信頼ポイントである公開鍵証明書への証明書パス検証を行うことで認証を行う。このディレクトリへの公開鍵証明書参照において、CAによる公開鍵証明書のディレクトリエントリへの格納と同様の照合が行われる。また、アプリケーションによっては、送信者の認証後に、公開鍵証明書内DNを用いてアプリケーション所有のデータベースなどとの照合を行う場合もある。例えば、Webサーバにおけるアクセス制御などがこれにあたる。

2 PKIにおけるDN照合規則

PKIにおける照合場面においてはある統一した照合規則が必要となる。

このPKIにおける照合規則として、RFC2459[1]において表1に示す2つの照合規則が規定されている。RFC2459照合は、本来RFC2459で推奨されている照合規則であるが、RFC459ではX.500シリーズとの互換性を保つため、X.500照合も認めている。これに従い、CAおよびアプリケーションにおいては、2つの照合規則に対応した照合機能を作りこむことで、これらの照合規則による照合を可能としている。

表1　RFC2459規定の2種類の照合規則

	X.500照合	RFC2459照合
String区別	なし	あり
大小文字区別	なし	なし(Printable) あり(その他)
スペース正規化	あり	あり(Printable) なし(その他)

しかし、一般に、ディレクトリは、これらの照合規則に対応した照合機能を持たない。そのため、RFC2253[2]において、これらの照合規則に対応するディレクトリエントリへのアクセス方法が規定されている。これに従い、CAによるディレクトリエントリの公開鍵証明書格納および利用者によるディレクトリエントリへの公開鍵証明書参照時には、X.500照合の時には“文字列”で、RFC2459照合を用いる時には“文字列（PrintableStringの場合）または16進数（PrintableString以外の場合）”でエントリ指定し、文字列に関する大文字/小文字区別，スペース正規化については、CAによるディレクトリエントリの格納時に行うことで2つの照合規則による照合を可能としている。

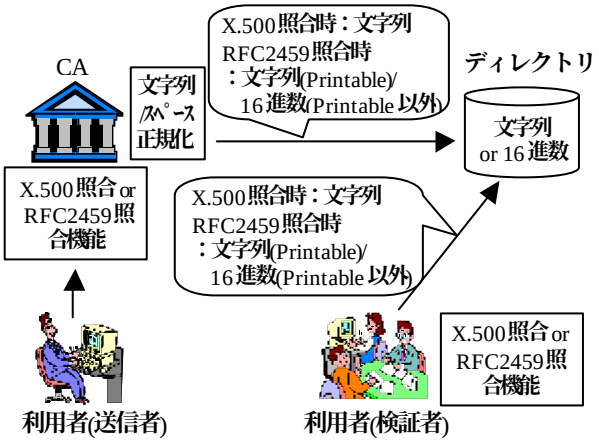

図2　各場面における照合方法

3 課題

2章で述べた照合規則および照合方法を現実社会(実際運用)に適用すると、以下の課題が生じる。

3.1　PKI適用時における照合方法の課題

ディレクトリへの公開鍵証明書参照，アプリケ

ーションにおける照合を行う利用者は、CA への公開鍵登録時に適用された照合規則を一般には把握していないと想定できる。そのため、以下の課題が生じる。

3.1.1 ディレクトリへの公開鍵証明書参照時課題

CA 時のエントリ格納方法と参照時のエントリ指定方法が一致するとは限らない。そのため、目的の公開鍵証明書をディレクトリから取得をすることができない場合がある。・・・課題1-1。

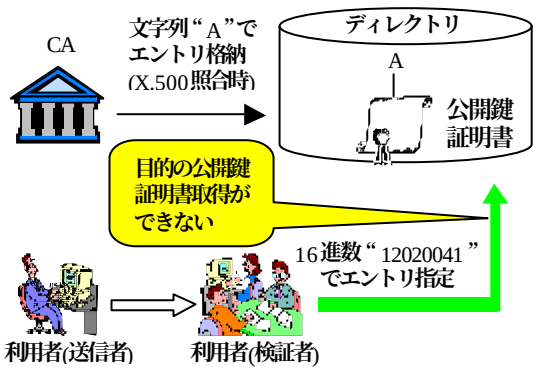

図3 ディレクトリからの公開鍵証明書取得不可①

例えば、CA において X.500 照合が行われており、“DN=（UTF8String）A”を持つ公開鍵証明書が発行された場合、2章で述べたように X.500 照合においてはディレクトリには文字列で格納されるため、この公開鍵証明書のエントリは文字列“A”で格納される。

しかし、検証者は CA への公開鍵登録時にどの照合規則で証明書が発行されているかが不明であるため、RFC2459 照合に対応したディレクトリエントリへのアクセスを行い、（PrintableString 以外は 16 進数でエントリ指定することにより）“12020041(UTF8String の‘A’を表す 16 進数)”でディレクトリアクセスを行った場合、目的の公開鍵証明書をディレクトリから取得をすることができない。

3.1.2 アプリケーションにおける照合時の課題

CA への公開鍵登録時に適用された照合規則とアプリケーションにおける照合で使用された照合規則も一致するとは限らない。そのため、本来許可すべきでない人も許可してしまう場合がある。・・・課題1-2

例えば、CA で RFC2459 照合が行われており、“DN=A”を持つ公開鍵証明書の利用者と“DN=a”を持つ公開鍵証明書の利用者は別人として登録されている。また、あるアクセス制御を行うアプリケーションがあり、データベースなどに“DN=A”の利用者をアクセス許可すると登録されていたとする。この場合において、“DN=a”を持つ利用者がこのアプリケーションを持つ利用者（検証者）へアクセスした際、検証者において X.500 照合により照合を行うと、本来アクセス権限をもっていないはずの“DN=a”を持つ利用者も許可してしまうことになる。

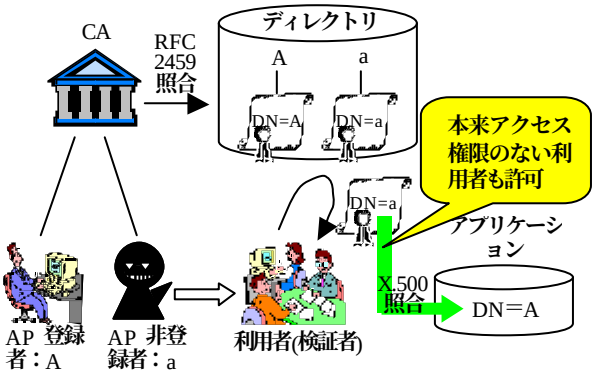

図4 アプリケーションにおける照合ミス

3.2 照合対象とする DN（名前）自身の課題

照合規則に関して、PKI の世界では X.500 照合と RFC2459 照合のみの規定であり、そこでは2章で述べたように“String 区別”“大小文字区別”“スペース正規化”の3つの照合しか行っていない。そのために以下の課題が生じる。

3.2.1 日本語特有のゆれの扱いに関する課題

日本語のかな表記には濁点や長音に関して日本語特有のゆれが存在しており、例えば“エヌティティ”と“エヌティーティ”や“なかた”と“なかだ”は同じ人として扱いたい場合も多くある[3]。しかし、既存の照合規則である X.500 照合と RFC2459 照合では別な人の証明書として解釈される。・・・課題2

3.2.2　別名（略称など）の扱いに関する課題

世の中には企業や協会の略称が認められている場合もある。そのため、“Nippon Telegraph and Telephone Company”は“NTT”という別名も持つ。しかし、既存の照合規則である X.500 と RFC2459 ではこれら２つの証明書は別な人として解釈されてしまう。

これを解決する手段として X.509[４]や RFC2459 には公開鍵証明書に格納する Subject の代替名称 Extension として SubjectAltName が規定されている。SubjectAltName にはインターネット電子メールアドレスや DNS 名，IP アドレスおよび別 DN などが標準規約として規定されている。上記を例にとれば、“Nippon Telegraph and Telephone Company”の SubjectAltName には“NTT”が格納される。

しかし、SubjectAltName に Subject の代替名称を記述したとしても、ディレクトリに公開鍵証明書を格納する際には、通常 Subject に記載されたエントリのみに格納する。そのため、以下の課題が生じる。

・Subject 名“A”，かつ代替名称“B”を持つ利用者から署名文書のみが送られた際、検証者は送信者の同一性を示す情報をもとに送信者の公開鍵証明書をディレクトリから参照するが、通信相手の DN が必ずしもその情報にある可能性は無く、目的の公開鍵証明書を取得できない可能性がある。・・・課題３

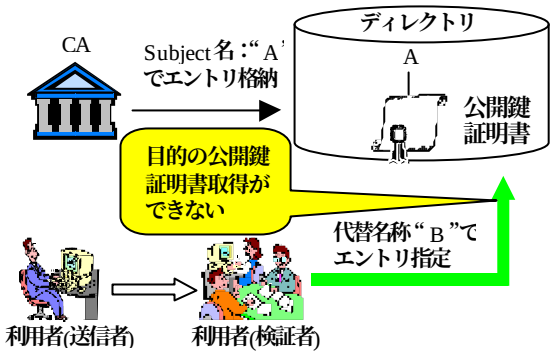

図５　ディレクトリからの公開鍵証明書取得不可②

上記課題を解決する策として、送信者が“自身の公開鍵証明書を署名付き文書とともに送る”あるいは“自身の公開鍵証明書に関する情報（自身の DN など）を送る”方法もあるが、すべてのアプリケーションが上記情報を送るわけではない。

4. 対策

本章においては３章における課題を解決する手段として、以下を提案する。

・課題１-１＆課題１-２に対して
　CA が管理・発行する資源であるディレクトリの特定エントリに照合規則を格納（利用者に公開）し、利用者はこの照合規則を利用してディレクトリアクセスおよびアプリケーションにおける照合を行う
・課題２に対して
　照合規則に日本語特有のゆれを含んだ照合規則など新規照合規則も適用可能とする
・課題３に対して
　公開鍵証明書に記載された同一の利用者を示すすべての名前のエントリに公開鍵証明書を格納し、検証者は送信者の同一性情報をもとにディレクトリアクセスする

4.1　実現手段

上記対策を用いた場合の処理フローを４.１.２に示す。なお、上記対策を実現するにあたり、必要となる照合規則を格納するディレクトリ属性型を４.１.１にて新規定義する。

4.1.1　新規ディレクトリ属性型定義

本方式においては、RFC2252[５]に規定されるディレクトリ属性型の他、以下のディレクトリ属性型を新規に定義して使用する。

属性型：NameComparisonType
属性値：CHOICE{ X.500, RFC2459,
　　J-X.500,J-RFC2459,・・・}

NameComparisonType は CA への公開鍵登録時にその CA が証明書を発行する上で採用している照合規則を格納する属性型であり、その属性値として、X.500 照合，RFC2459 照合の他、新規照合規則も追加可能とする。

なお、J-X.500，J-RFC2459 は、新規照合規則

の例として、日本語特有のゆれの正規化照合を行う照合規則を追加したものである。

4.1.2　処理フロー

本方式は以下の方式により実現される。

1）CA 側処理

手順１：

CA 構築時の各種設定および運用開始後のポリシー変更において、ディレクトリ NameComparisonType 属性型に照合規則情報を格納する。

手順２：

利用者からの申請に基づく公開鍵証明書発行時、CA は利用者が既に登録している人か、新規登録者かの確認を手順１で設定した照合規則で行うとともに、同一の利用者を示す Subject 名以外の Name を公開鍵証明書の SubjectAltName に格納する。

手順３：

公開鍵証明書のディレクトリへの格納時に、Subject のエントリだけでなく SubjectAltName に記述された Name のエントリにも公開鍵証明書を格納する。

例えば、図６に示すように、CA において J-X.500 照合を行っている時には、ディレクトリ NameComparisonType 属性型に“ J-X.500 ”を格納する（手順１）。そして、利用者からの公開鍵登録申請時に J-X.500 照合により申請された利用者の確認を行うとともに、公開鍵証明書に“ Nippon Telegraph and Telephone Company ”の代替名称として“ NTT ”を SubjectAltName に格納する（手順２）。その後、手順１にて J-X.500 を格納したディレクトリの“ Nippon Telegraph and Telephone Company ”エントリと“ NTT ”エントリに対して、それらの Name を持つ公開鍵証明書を格納する（手順３）。

2）利用者側処理

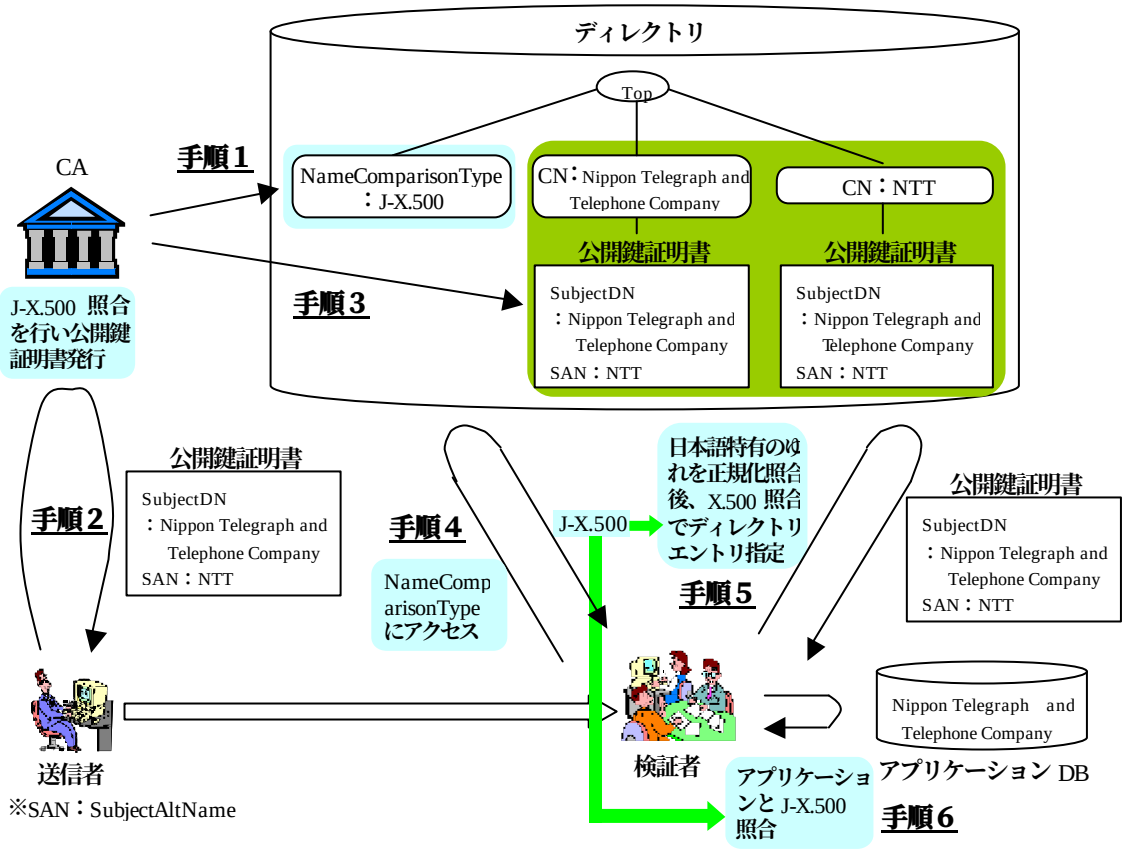

図６　実現手段

手順４：

送信者からの署名文書受取り時に、ディレクトリのNameComparisonType属性型にアクセスし、照合規則を取得する。

手順５：

手順４で取得した照合規則に即したディレクトリエントリ指定方法でディレクトリアクセスを行い、目的の公開鍵証明書を取得する。なお、手順４で取得した照合規則がX.500照合およびRFC2459照合以外であった場合、その照合規則に対応した正規化を行った後、X.500照合あるいはRFC2459照合に対応したディレクトリエントリ指定方法でディレクトリアクセスを行う。

手順６：

手順４で取得した照合規則に従い、アプリケーションにおけるとの照合を行う。

図６の例を用いると、送信者“Nippon Telegraph and Telephone Company”から署名文書を受け取った検証者はディレクトリのNameComparisonType属性型にアクセスし照合規則“J-X.500”を取得する（手順４）。そして、手順４にて取得した照合規則は“J-X.500”であるため、[３]に示された照合規則により日本語特有のゆれを正規化照合後、X.500照合に対応したエントリ指定方法でディレクトリアクセスすることで送信者“Nippon Telegraph and Telephone Company”のディジタル署名を検証する公開鍵証明書を取得する。なお、この時の検証者によるエントリ指定は“Nippon Telegraph and Telephone Company”だけでなく“NTT”でも目的の公開鍵証明書が取得可能である（手順５）。送信者“Nippon Telegraph and Telephone Company”の認証後、手順４で取得した照合規則“J-X.500”により、検証者が所有するデータベースに登録されたNameと送信者の公開鍵証明書に格納されたName（Subject名あるいはSubjectAltName）との照合を行う（手順６）。

5. おわりに

本稿では、PKIで規定されている照合規則および照合方法を実装して現実社会に適用する際の課題および対策について述べた。

今後、本稿で述べた方式の実装評価を行い、その実現性について評価を行っていく予定である。

なお、PKIにおける照合規則の課題を述べるにあたり、本稿ではPKI特有の課題および対策について、考察を行ったが、現実社会における照合の潜在的問題として以下の課題についても考慮する必要があることも忘れてはならない。

１）同姓同名に関する問題

同姓同名の人が存在することによる人を一意に特定できない可能性。現実社会においても同姓同名の犯罪が起こっており、この対策として、息子という言葉をつけて区別したり、アメリカやカナダではSIN(Social Insurance Number)で管理している。

２）名前の別人による再利用に関する問題

過去に使用していた名前を他の人が再利用する可能性。

３）良く似た文字の利用に関する問題

よく似た名前を登録しておくことにより、成りすましを起こさせる可能性。例えば、慎重でないユーザが存在していた場合（これがほとんどであるが）、現実社会においては“YahOO”のDNにも関わらず“Yahoo”と誤認識してしまう可能性もある。“１”と“l”も同じことが言える。

【参考文献】

[１]RFC2459(Certificate and CRL Profile) IETF PKIX-WG(1999)

[２] RFC2253(Lightweight Directory Access Protocol (v3):UTF-8 String Representation of Distinguished Names)(1997)

[３] 人名のかな表記のゆれに基づく近似文字列照合法 高橋克巳，梅村恭司　情報処理学会ジャーナル Vol.36 No.08(1995)

[４] ITU-T Recommendation X.509(1997E)

[５] RFC2252(Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions)(1997)