秘密鍵証明書・属性証明書を利用した暗号電子メールシステム

鮫島 吉喜　　宮崎 博
same@ori.hitachi-sk.co.jp　　zaki@ori.hitachi-sk.co.jp
日立ソフトウェアエンジニアリング(株) 研究部

平成7年10月26日

概　要

秘密鍵証明書と属性証明書及び認証、復号サーバを導入して、暗号メールを実現した。特徴として次がある: (1) ユーザ識別子ではなく、肩書、所属のようなユーザ属性や時刻といったコンテキスト情報を用いて、メール受信人の指定や発信人の認証が可能である。(2) サーバはユーザの秘密鍵や属性情報を保持しないので、サーバへの攻撃を減らすことができる。本発表では、暗号メールのプロトコル、実装について報告、考察する。

1　はじめに

1.1　動機と目的

インターネットの急速な普及にともない、ネットワーク上でビジネスを展開する動きが盛んである [19]。このため、秘匿、認証、課金などのセキュリティ技術が注目されており、低レイヤの IP/Secure[17]、Secure TCP[18] を始め、遠隔地端末やファイル転送などの既存アプリケーションのセキュリティ対応が進んでいる [1][11]。中でも広く普及している電子メールや World Wide Web[12] は、ネットワークビジネスの有力な手段となるので、セキュリティを強化したプロトコルの標準化や実装が進んでいる [10][5]。

これらのプロトコルやアプリケーションは、公開鍵暗号方式と秘密鍵暗号方式を組み合わせて、通信相手の認証、通信データやメッセージの秘匿、改竄検知機能を実現している。

秘密鍵暗号方式は、発信者と受信者が同じ鍵を共有してデータを暗号、復号するものであり、米国で標準化されている Data Encryption Standard (DES) や日立製作所が開発した MULTI[16] がこの方式の暗号アルゴリズムである。

これに対して公開鍵暗号方式は、暗号と復号に使う鍵が異なっており、各ユーザが個人鍵と公開鍵のペアを持っている。個人鍵は所有ユーザが他ユーザからアクセスできないように保管、使用する鍵であるのに対して、公開鍵は名前のとおり通信相手全てに公開、配布する鍵である。公開鍵暗号方式は元になるアイデアが 1970 年代に発表され [3]、米国ばかりでなく日本でも特許が成立している。このためアルゴリズムの使用に際して権利者から使用許可を受けなければならない [4]。

筆者らは公開鍵暗号方式を使用せず、秘

Privacy Enhanced Message System using Secret-Key and User-Attribute Certificates.
Yoshiki SAMESHIMA, Hiroshi MIYAZAKI
Research & Development Department,Hitachi Software Engineering Co., Ltd.

密鍵証明書と属性証明書を導入し、メッセージの代理受信やユーザ属性(ユーザ情報)の認証が可能な暗号電子メールプロトコルについて検討した。本論文では、既存電子メールシステムの問題点、二つの証明書、検討した暗号電子メールプロトコル、及び実装について報告、考察する。

2 既存暗号メールの問題点

2.1 公開鍵暗号利用上の問題点

広域ネットワークでの認証や秘匿機能の実現にはRSAアルゴリズム[14]を代表とする公開鍵暗号方式の使用が一般的である。これは秘密鍵暗号方式ではユーザのペアごとに鍵が必要なのに対して、公開鍵暗号方式では各ユーザが一対の鍵を持っていれば良く、大規模ネットワークでの鍵管理がしやすいためである。しかし、米国では公開鍵暗号方式の使用についていくつかの制限がある[4]。

まず、一暗号アルゴリズムであるRSAを含め、公開鍵暗号方式自体が特許であり、ハードウェア、ソフトウェアを問わず、利用や製品化にはPublic Key Partnersのライセンスが必要である。Apple、DEC、Lotus、Microsoft、Sunなどがライセンスを受けている。日本でも同様の特許が成立しており、状況は同じと考えられる。

次に、米国からの輸出規制がある。暗号が認証機能に使われる限りは緩やかであるが、秘匿機能に使う場合には、国外のアメリカ企業で使用するケースを除いて、一般に禁止されている。このため米国で実績のあるシステムやライブラリを使う自由が制限されたり、開発に際して相互接続性を確認するのに不自由が生じる。これは秘密鍵暗号方式についても同様であり、鍵の長さが限定され暗号強度が弱い場合だけに輸出が許可されている。

2.2 広域ネットワークでの鍵配布と問題点

暗号を使うには、「正しい」鍵の入手、管理、使用が最も重要である。つまり、意図した通信相手との交信に使う正しい鍵を入手、使用しなければならない。正しい鍵を配布するためには、Key Distribution Centre (KDC)を使用するのが一般的である[13]。

公開鍵暗号方式のKDCであるCertification Authority (CA)は、ユーザ名と公開鍵を結びつける証明書を発行する静的なオフラインのKDCである[6]。これに対して秘密鍵暗号方式を採用した場合、KDCは常時アクセス可能である必要がある[9]。これは暗号、復号のたびに鍵の生成や配布などを、ユーザがKDCに依頼する必要があるからである。このためにKDCの複製(スレーブサーバ)を用意して可用性を高める方法がある。しかし、この方法ではユーザの秘密鍵を持ったKDCが多数存在することになり、さらにマスタサーバからスレーブサーバへの鍵の転送を含めて、攻撃の対象が増えることになる。このため鍵の転送やKDCをどう攻撃から守るかが問題となる。

2.3 発信者認証

文書に署名する場合には、単に名前だけでなく署名の日付や開発部部長のような肩書を加えるのが普通である。印影の大きさや色、形で役職を示す方法もある。一方、郵便には発信証明や内容証明といった、郵便局が発信や配信の事実を証明するサービスがある。

これに対して既存の暗号メールシステム、例えばPEM[10]の場合、認証対象はメッセージの中身とその発信者識別子だけであり、他の証明サービスはない。X.400 Message Handling System (MHS) [7]では、

Message Transfer Agent (MTA) が発信証明、配信証明などの書留機能を提供するが、証明内容は発信者と時刻、内容と固定的である。ここで次の事実に注意する必要がある。MHS では発信者、受信者に対して第三者である MTA が発信証明、配信証明をしているのに対して、文書への署名は肩書や日付という内容を発信者 (署名者) 自身が示しているだけで、何らかの authority が保証しているわけではない。

メッセージシステムが個人間の連絡だけでなく、組織間の契約や決済、組織内での決済や稟議などに使われる場合には、単に発信者の識別子がついた署名だけではなく、肩書のような発信者の属性や発信時刻などについて第三者や組織の管理部門による認証が必要になる。

2.4 代理受信

メッセージは個人だけに送られるものではない。例えば、開発部部長や仕入れ業務担当者のように、肩書や担当といった属性に対して送られるものもある。さらに、部長が不在の間、部長代理や部長秘書が許可を受けて部長宛のメッセージを読むことも必要である。

秘書に代理受信してもらうためには、部長は秘書に自分の鍵を渡す必要がある。しかし、一度鍵を渡してしまうと、秘書は部長が出張から帰った後でも、ネットワークをモニタすることで部長宛のメッセージを読むことができる。これは部長や担当者が交替した場合でも同様であり、代理受信や担当者交替のたびに鍵を交換する必要が生じる。決められた期間だけ、属性に応じた受信ができる仕組みが必要である。

2.5 暗号メーリングリスト

メーリングリストで暗号電子メールを使うには主に二つの方法がある。一つはリスト用の暗号鍵、復号鍵 (秘密鍵暗号方式なら同一) をメンバに配布し、メーリングリスト宛のメッセージはこれらの鍵で暗号、復号する方法である。他方は発信者はリストの展開サーバ宛に暗号して送り、サーバは復号の後、個々のメンバの暗号鍵で暗号し直して発信するという方法である。

最初の方法には脱会したメンバが、何らかの手段でリスト宛のメッセージを入手した場合に、自分が持っている復号鍵で復号できるという問題がある。これを解決するために脱会のある度に鍵を交換する方法もあるがコストがかかる。

第二の方法には本来受信人ではないリスト展開サーバがメッセージにアクセスするという問題点がある。

3 秘密鍵証明書と属性証明書

3.1 秘密鍵証明書

KDC への攻撃の対策として、秘密鍵証明書 [2] を導入する。これはユーザの秘密鍵を KDC が保持する代わりに、ユーザ識別子と KDC の鍵で暗号した秘密鍵のペアに KDC が署名した秘密鍵証明書を利用するものである。ユーザは KDC に認証、復号の依頼をする時点で、依頼と一緒にその証明書を送る。KDC は証明書の署名を確認、復号してユーザの秘密鍵を取り出し、依頼を処理することができる。秘密鍵自体は KDC の秘密鍵で暗号、さらに証明書は署名されているので、KDC 以外はユーザの秘密鍵にアクセスしたり、証明書を偽造することができない。このため証明書の蓄積や配布は自由であり、管理コストが安くなる。

3.2 属性証明書

送受信者の属性の問題を解決するために属性証明書を導入する。(ユーザ) 属性には氏名、所属部署、所属グループ、肩書、役割、担当業務や加入しているメーリングリ

スト識別子などがある。また、わかり易いユーザインターフェースを提供するために印影を導入することもできる。

属性は属性型と属性値からなる。例えば、「肩書 = 開発部部長」という属性は「肩書」という属性型と「開発部部長」という属性値からなる。

属性は人事や情報管理部門などの組織内のauthorityが発行するユーザ属性証明書に含まれる。証明書の中身は、ユーザ識別子、属性、有効期間、シリアル番号などである。証明書はKDCの秘密鍵で署名することで、偽造や改竄があっても、検知することができる。このため鍵証明書と同様に蓄積、転送は自由である。

4 暗号電子メールプロトコル

4.1 記号の説明

秘密鍵暗号方式、秘密鍵証明書、属性証明書を使用した暗号メールプロトコルを説明する前に、使用する記号を表1にあげる。

ここでコンテキスト属性とは、時刻や発信者のネットワーク位置など、メッセージの送受信の度に変わる動的な情報を言う。これに対して前章で述べた属性(ユーザ情報)は静的であり、変更が少ない。

証明書制御情報には有効期間やシリアル番号などがある。証明書の署名が正しくても、有効期間が過ぎた証明書は無効になる。シリアル番号は無効証明書リストに使われる。つまり、有効期間内に鍵を変更したり、異動して所属部署というユーザ属性が変わった場合など、証明書の内容が無効になった時には、無効証明書リスト(ブラックリスト)にシリアル番号を載せる。証明書を使用する際には、署名を確認し、有効期間内であること、無効リストに載っていないことを確かめた後に、中身の情報を利用する。

表1: 記号表

#	記号	意味
1	S	KDC
2	X	発信者
3	Y	受信者
4	A_i	Xのユーザ属性 $(i=1,2,\ldots,m)$
5	B_j	Yのユーザ属性 $(j=1,2,\ldots,n)$
6	C_k	コンテキスト属性 $(k=1,2,\ldots,l)$
7	C'_k	C_kの属性型
8	K_Z	Zの秘密鍵
9	MSG	メッセージ本体
10	k	MSGを暗号する秘密鍵
11	h	MSGのメッセージダイジェスト
12	L	証明書制御情報
13	$\{I\}_K$	秘密鍵Kを使い暗号したデータI
14	$\{I\}^K$	秘密鍵Kを使い署名したデータI
15	$\{Z,\{K_Z\}_{Ks},L\}^{Ks}$	Sが発行したZの秘密鍵証明書
16	$\{X,A_i,L\}^{Ks}$	Sが発行したXの属性証明書

4.2 プロトコル

KDC Sから発信人X、受信人Yに証明書を発行した場合のXとS、XとY、YとS間のプロトコルを述べる。

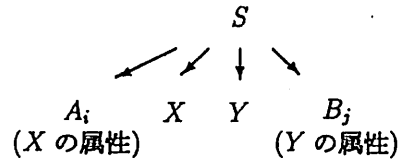

図1: KDC Sから証明書を発行

図1は証明書の発行関係を示す。ここでは、SはXとYの鍵証明書、XのA_i属

性証明書、Y の B_j 属性証明書を発行している。

4.2.1 認証依頼

$$X \rightarrow S: \quad \{k, h, B_j, A_i, C'_k, X, S\}_{K_X},\ \{X, \{K_X\}_{K_S}, L\}^{K_S},\ \{X, A_i, L\}^{K_S}$$

ユーザ X がメッセージ MSG を作成する。MSG の暗号鍵 k をランダムに生成、MSG のメッセージダイジェスト h を求める。

受信者の属性 $B_1, \ldots, B_n$、k、h と KDC から認証して欲しい属性 $A_1, \ldots, A_m$、コンテキスト属性型 $C'_1, \ldots, C'_l$、X、S を自分の秘密鍵 K_X で暗号、$\{k, h, \{B_1, \ldots, B_n\}, \{A_1, \ldots, A_m\}, \{C'_1, \ldots, C'_l\}, X, S\}_{K_X}$ を生成する。ここで $\{B_1, \ldots, B_n\}$ にコンテキスト属性が含まれていても構わない。この時は受信人が復号できる時刻やネットワーク位置を指定したことになる。これに A_i 属性証明書と鍵証明書を添付して S に送り、認証を依頼する。以降、本報告では、$\{A_1, \ldots, A_m\}$ を A_i、$\{B_1, \ldots, B_n\}$ を B_j、$\{C'_1, \ldots, C'_l\}$ を C'_k と略記する。

4.2.2 認証応答

$$S \rightarrow X: \quad \{\{k, h, B_j, A_i, C_k, X, S\}_{K_S}, k, h, X, S\}_{K_X}$$

S は鍵証明書を自分の秘密鍵を使って確認、$\{K_X\}_{K_S}$ を復号して X の鍵 K_X を得る。次に X の認証依頼を K_X で復号、X から自身 S 宛に来た依頼であることを確かめる。

次に属性名 C'_j からコンテキスト属性 C_j を生成する。例えば C'_1 が署名時刻の場合、KDC が自分の時計を元に、動的にコンテキスト属性、「署名時刻 = 951026091034」(1995 年 10 月 26 日 9 時 10 分 34 秒) を 生成する。ユーザ属性に関しては、依頼の中の A_i に対応する属性証明書 $\{X, A_i, L\}^{K_S}$ を確認し、X が属性 A_i を持っていることを確かめる。

k、h、B_j、A_i、C_k、確認した認証依頼元 X と自身 S を自身の鍵 K_S で暗号、認証情報 $\{k, h, B_j, A_i, C_k, X, S\}_{K_S}$ を生成する。

S から X への返答に対するリプレイ攻撃を防止するために、k、h、X、S を含めて認証情報を X の鍵 K_X で暗号、返答を生成して、X に送る。

4.2.3 メッセージ発信

$$X \rightarrow Y: \quad \{MSG\}_k,\ \{k, h, B_j, A_i, C_k, X, S\}_{K_S},\ B_j$$

X は S からの返答を自分の鍵 K_X で復号、k、h が自分が要求したものであることと、返答の送り先が自身 X、送り元が要求先の S であることを確かめる。この後、暗号したメッセージ $\{MSG\}_k$ と認証情報、受信人のユーザ属性 B_j を Y に送る。

4.2.4 復号依頼

$$Y \rightarrow S: \quad \{k, h, B_j, A_i, C_k, X, S\}_{K_S},\ \{Y, B_j, L\}^{K_S},\ \{Y, \{K_Y\}_{K_S}, L\}^{K_S}$$

メッセージを受信した Y は、K_S で暗号された認証情報を利用できないので、S に自分の鍵 K_Y で暗号し直してもらう。X が送った B_j から受信人のユーザ属性がわかるので、対応する自分の属性証明書を添えて依頼する。また B_j にコンテキスト属性が含まれている場合には、現在のコンテキスト、例えば時刻が B_j に一致しているかを確かめて依頼する。

4.2.5 復号応答

$$S \rightarrow Y: \quad \{\{k, h, B_j, A_i, C_k, X, S\}_{K_S}, k, h, A_i, C_k, X, S\}_{K_Y}$$

S は認証情報を K_S で復号する。ここで受信人は属性 B_j をもつユーザであり、認証情報は自身 S が生成したことがわかる。

Yの属性証明を確認することで、Yが属性B_jを持っており、Yが正規受信人であることがわかる。B_jがコンテキスト属性だった場合には、現在のコンテキストに合致するかも確認する。Yの鍵証明書を確認、K_Yを取り出し、これを使って認証情報の内容を暗号して確認情報を生成、Yに返す。この時、リプレイ攻撃防止のため元の認証情報を含める。

4.2.6 メッセージ確認

Yは自分の鍵K_Yを使ってSからの返答を復号、自分が依頼したSの認証情報が含まれていることを確認する。MSGの鍵k、メッセージダイジェストh、発信者Xの属性A_i、Xが認証を要求した時のコンテキスト属性C_k、発信者Xと、どのKDCが認証したかの情報Sを得る。kを使って暗号メッセージ$\{MSG\}_k$を復号、MSGを得る。MSGのメッセージダイジェストを計算、hと比較して改竄がなかったかどうかを確かめる。

5 実装

上記プロトコルにしたがった暗号電子メールシステムをWindows™ [1]上に実装した。秘密鍵暗号アルゴリズムには日立製作所が開発したMULTI、メッセージダイジェストにはMD5[15]を使用した。暗号したメッセージの転送には既存の電子メールシステムを採用した。

6 結果及び結果の検討

6.1 基本サービスの実現

これまで示したように、公開鍵暗号方式を用いず、秘密鍵暗号方式のみを利用してメッセージの発信者認証、改竄検出、秘匿サービスを実現した。

[1]Windowsは Microsoftの登録商標です。

6.2 KDC管理負担の軽減

Kerberosサーバ[9]のようにKDCがユーザの秘密鍵を保管する必要がないので、サーバの安全管理負担を軽減することができた。これはユーザ属性についても同様である。しかし、サーバは自身の秘密鍵を保持しているので、それをいかに保護するかの問題は残る。

6.3 発信と発信者の属性認証

発信や発信者に関して、ネットワーク位置や時刻などのコンテキスト属性、肩書や担当などのユーザ属性をKDCが証明することができた。一組織内であれば、人事や情報管理などの部門が証明書を発行、KDCを運用して信頼性が確保できる。

認証依頼において、コンテキスト属性型ではなくメッセージ名や文書フォーマットなどの属性を送ることで、メッセージの付加情報(属性)を認証することもできる。この場合、KDCが認証するのではなく発信者が認証した情報となる。

6.4 属性依存の復号

対応する属性証明書を提示することで、役割や担当宛に来たメッセージを復号することができる。証明書の中の有効期間を調整することで、例えば、部長の出張中だけ部長宛のメッセージを代理人が読むことができる。

6.5 メーリングリストへの応用

加入していることを示すユーザ属性を用いることで、メーリングリストに適応できる。脱会者が出ても、そのユーザに発行した属性証明書を無効にすることで、既存の第一の方法にあった問題を解決できる。既存の第二の方法に沿った場合、つまりメーリングリスト展開サーバとKDCが同一で

ある場合は未解決のままである。展開サーバとKDCの管理者を分けて運用する必要がある。

6.6 証明書への攻撃

証明書はサーバの秘密鍵で署名されている。このため証明書の偽造は不可能であり、自由に配布、保管できるので管理コストが小さい。しかし、KDCの秘密鍵に対する既知平文攻撃、場合によっては選択平文攻撃が可能となる。サーバの秘密鍵の長さを大きくしたり、より安全な暗号アルゴリズムを使うなど、証明書の署名には、特に注意が必要である。

6.7 広域ネットワークへの適用

本論文では単一ドメイン、単一KDCでのプロトコルを示した。広域ネットワークに適用するためには複数ドメイン、複数KDC間のプロトコルを導入しなければならない。Kerberosのように、各KDC間に使われる秘密鍵を定め、認証情報をやりとりしたり、KDCを階層化して上位のKDCが認証や復号を行う方法も考えられる。しかし、インターネットのような世界規模のネットワークへの適用には無理があり、公開鍵暗号方式との併用は避けられないと考えている。

6.8 鍵の保管

現在の実装ではユーザの秘密鍵をユーザパスワードを使って暗号して、フロッピーディスクに格納している。最近はCPUを持ち、本体内で暗号、復号を実現し、秘密鍵自身は本体の外に出ないICカード(スマートカード)も市販されている。単にプロトコルの安全性に注意を払うだけでなく、実用のためにはサーバとユーザの秘密鍵の保管、使用などの運用面での安全性の検討をする必要がある。

6.9 認証、復号サービスの分離

現在のプロトコルでは、属性証明書さえあれば認証(署名)も復号も可能となる。代理として、部長宛の暗号メールの復号はできるが部長としての署名はできないなど、認証と復号サービスを分離した方が良いと思われる。さらにメッセージの重要度や機密度に応じて、鍵や暗号アルゴリズムを使い分ける必要がある。

7 結論

秘密鍵証明書と属性証明書を導入した暗号電子メールシステムを報告した。特徴として、ユーザ識別子だけでなくユーザ属性やコンテキスト属性を用いた受信者指定、発信者認証が可能である。また、証明書を導入したことで認証/復号サーバがユーザ秘密鍵やユーザ属性を保持する必要がなく、サーバの管理負担を減らすことができた。

今後、スマートカード導入したサーバやユーザ鍵の保管、使用、認証と復号依頼に使えるユーザ属性もしくは属性証明書の使いわけなどを検討していく。また本プロトコルは電子メールだけでなく、クライアントサーバシステムなど他の通信システムにも適応可能であり、適用分野を広げていく予定である。

参考文献

[1] D.Borman, *Telnet Authentication Option*, Internet RFC 1416, Internet Activities Board (1993)

[2] D.Davis, R.Swick, *Network Security via Private-Key Certificates*, OS Review, 24, 4,(1990)

[3] W.Diffie, M.E.Hellman, *New directions in cryptography*, IEEE Transactions on Information Theory, pp. 644-654 (1976)

[4] P.Fahn, *Answers To FREQUENTLY ASKED QUESTIONS About Today's Cryptography, version 2.0*, RSA Laboratories (1993)

[5] K.Hickman, *The SSL Protocol*, Internet Draft, Internet Activities Board (1995)

[6] International Standards Organization, *Information Processing – Open Systems Interconnection – The Directory – Authentication Framework, International Standard 9594*, (1988)

[7] International Standards Organization, *Information Processing Systems – Text Communication – Message Oriented Text Interchange System – part 4: Message Transfer System: Abstract Service Definition and Procedures, International Standard 10021-4*, (1988)

[8] S.Kent, *Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management*, Internet RFC 1422, Internet Activities Board (1993)

[9] J.Kohl, *The Kerberos Network Authentication Service (V5)*, Internet RFC 1510, Internet Activities Board (1993)

[10] J.Linn, *Privacy Enhancement for Internet Electronic Mail: Part I: Message Encryption and Authentication Procedures*, Internet RFC 1421, Internet Activities Board (1993)

[11] S.J.Lunt, *FTP Security Extensions*, Internet Draft, Internet Activities Board (1995)

[12] M.Marriott, A.Underwood, *Super Cyber Surfers - The Web: How to get around the most fun place on the Internet*, Newsweek, pp. 43-45 (20th March 1995)

[13] R.M.Needham, M.D.Schroeder, *Using Encryption for Authentication in Large Networks of Computers*, Communications of the ACM, 21, 12, pp. 993-999 (1978)

[14] RSA Data Security, Inc., *Public-Key Cryptography Standards #1: RSA Encryption Standard*, RSA Data Security, Inc. (1993)

[15] R.Rivest, *The MD5 Message-Digest Algorithm*, Internet RFC 1321, Internet Activities Board (1992)

[16] 宝木 他, マルチメディア向け高速暗号アルゴリズム Hisecurity-Multi2 の開発と利用方法, 1989 年 情報理論とその応用 暗号と情報セキュリティジョイントワークショップ資料, 電子情報通信学会, pp.167-173 (1989)

[17] T.Tanida, Y.Shinoda, *IP/Secure: Providing security on datagram delivery for mobile host environment*, Proceedings of INET'94/JENC5, pp. 643-661 (1994)

[18] T.Tsutsumi, S.Yamaguchi, *Secure TCP – providing security functions in TCP layer*, Proceedings of the INET95 pp. 905-913 (1995)

[19] J.W.Verity, R.D.Hof, *THE INTERNET: How it will change the way you do business*, Business Week, pp. 38-46 (14th November 1994)