6Y-7

TPMに基づく端末認証のための認証局の構築

大川雅士† 篠田昭人‡ 脇田知彦‡ 福田洋治†† 毛利公美† 白石善明‡ 野口亮司‡‡
岐阜大学† 名古屋工業大学‡ 愛知教育大学†† (株)豊通シスコム‡‡

1. はじめに

企業などの組織では，ユーザ認証によるアクセス制御は一般的に行われている．一方で，ノートPCやスマートフォンなどの端末が安価に入手できるようになり，ユーザ認証だけでは正当な利用者による組織が把握していない端末からの情報システムへのアクセスが許されることになる．そのような端末からの組織内の機密情報を漏洩させないためには，端末の特定をした後にアクセスを許可するような端末認証が要素技術として有用である．

端末認証にはセキュリティチップと呼ばれる耐タンパー性を有するICチップが使える．その1つである，PCのマザーボードに実装されているTPM（Trusted Platform Module）はRSA暗号の演算・鍵生成・格納や，乱数生成・ハッシュ演算・ハッシュ値保管などの端末認証を安全に行うのに必要な機能が搭載されている[1].

TPMを一意に特定できる秘密鍵と対をなす公開鍵に対して，信頼できる認証局が公開鍵証明書を発行し，その公開鍵証明書を用いれば端末認証ができる．つまり，端末認証には認証局の構築が必要となる．

TPMのための認証局の参照実装[2]がすでに存在する．ここでは，認証局が動作しており，端末側の証明書発行のLinux向けのサンプルコードが公開されている．しかし，企業で用いられているPCの大半はWindowsを搭載していることから[3]，Windowsの端末認証と企業などの組織を想定した認証局の参照実装を本研究では目標としている．

2. TPMと端末認証

2.1. TPMとIdentity鍵

TPMとはTrusted Computing Group（TCG）が策定した仕様に基づき，PCのマザーボードに実装されているセキュリティチップのことである．初期出荷時にPCとチップ間で紐付けられた情報があり，他のPCにTPMを移して利用することが不可能な耐タンパー性を有するという特徴を持つことから，より厳密な認証ができる．

端末を一意に特定するためのTPMの鍵として，Identity鍵（AIK：Attestation Identity Key）をTPM所有者が生成できる．Identity鍵とはRSA鍵ペアであり，以降では，RSAの秘密鍵と公開鍵に対応するIdentity鍵をそれぞれIdentity秘密鍵，Identity公開鍵と呼ぶ．

2.2. Identity鍵による端末認証

文献[1]に示されているPKI（Public Key Infrastructure）をベースに考えられた端末認証は図1のようになり，次のような流れになる．まず，(1) PCの初期出荷時にTPMに格納されているEK（Endorsement Key）と呼ばれる鍵を用いて端末の正当性検証に用いるIdentity鍵を生成する．(2) EK証明書，Identity公開鍵，Identity秘密鍵による署名などをプライバシーCAに送付する．そして，プライバシーCAは，(3) EKが正しいことを検証し，(4) Identity鍵証明書を作成する．その後，(5) Identity鍵証明書を端末に発行し，(6) 端末認証サーバに端末情報を登録する．端末が認証を受ける際には，(7) Identity鍵証明書を端末認証サーバに送付して検証される．

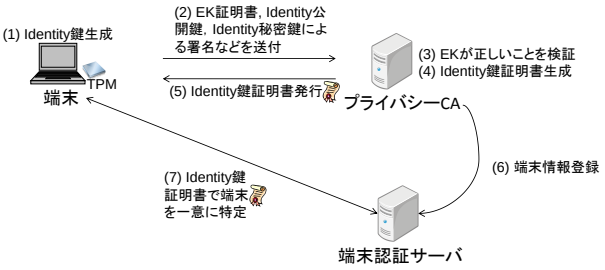

図1 Identity鍵証明書発行と端末認証の流れ

2.3. PKIと端末認証のための認証局

文献[1]の端末認証はPKIの仕組みに基づいている．PKIを構成する主な構成要素には(1) 認証局（Certification Authority，CA），(2) 登録局（Registration Authority，RA），(3) リポジトリ，(4) アーカイブ，(5) 証明書所有者，(6) 証明書利用者の6つがあり，(1)~(4)をPKIコンポーネント，(5)~(6)をPKIユーザとして[4]では分類している．CAは，証明書所有者に証明書を発行，証明書を失効させて証明書失効リスト（Certificate Revocation List，CRL）を発行し，証明書とCRLをリポジトリに公開する．RAは，PKIを大規模で運用する際の発行権限の分散管理を可能とするために，本人性の確認を行う，CAに対して証明書の発行や失効を要求する．

企業での端末認証を想定すると，TPM搭載端末と利用者の紐付けは重要であり，特に，CAの管理者がいる場所とは地理的に分散しているオフィスがある場合には，RAを導入した認証局を構築し，本人性確認の手続きは分散できる構成をとるのが望ましいと言える．

3. プライバシーCAの設計

3.1. Identity鍵証明書発行要求の処理手順

TPMの仕様書[5]に従うと，利用者が端末のIdentity鍵証明書を発行してもらうためには，TPM_IDENTITY_REQと呼ばれる証明書発行要求のためのデータを生成してCAに送信し，CAからasymCaContentsBlob, symCaAttestationBlobと呼ばれるデータを受け取る．発行側となるCAは図2に示した次の処理を行う．

A Design of Privacy Certification Authority for TPM-based Terminal Authentication

† Masashi Ohkawa and Masami Mohri・Gifu University
‡ Tomohiko Wakita, Akihito Shinoda and Yoshiaki Shiraishi・Nagoya Institute of Technology
†† Youji Fukuta・Aichi University of Education
‡‡ Ryoji Noguchi・Toyotsu Syscom Corp.

(1) TPM_IDENTITY_REQ から，TPM_IDENTITY_PROOF という端末で生成されたセッション鍵で暗号化された Identity 公開鍵が含まれている symBlob というデータと，その暗号化されたセッション鍵が含まれている asymBlob というデータを取り出す.
(2) asymBlob を CA 秘密鍵で復号し，セッション鍵を取り出す.
(3) セッション鍵で symBlob を復号し，TPM_IDENTITY_PROOF を取り出す.
(4) TPM_IDENTITY_PROOF から EK 証明書を取り出し，EK が正しいことを検証する.
(5) TPM_IDENTITY_PROOF から Identity 公開鍵を取り出し，ハッシュ関数によりダイジェストを作成し，CA 秘密鍵で署名した Identity 鍵証明書を作成する.
(6) セッション鍵を生成し，Identity 鍵証明書を暗号化し，TPM_SYM_CA_ATTESTATION とする．これが symCaAttestationBlob となる.
(7) Identity 公開鍵をハッシュ関数に入力し，ダイジェストを作成し，Identity 鍵証明書を暗号化したセッション鍵とともに TPM_ASYM_CA_CONTENTS とする．これを EK 証明書から取り出した EK 公開鍵で暗号化したものが asymCaContentsBlob となる.
以上の処理が CA に行われた後，端末側にデータが送られ，Identity 証明書に基づいて Identity 鍵がアクティベートされる.

3.2. Identity 鍵証明書発行の手順

TPM の仕様書には，本人性確認と端末と Identity 鍵の関連付けについての具体的な手順は述べられていない．2.3 節で述べたように，プライバシーCA の構築にあたっては，RA の存在を考慮した発行手順を設計する必要がある.

端末（証明書所有者），RA，CA は図 3 に示した次の手順で Identity 鍵証明書を発行する.
(1) 【端末】本人（利用者）確認要求の送信
(2) 【RA】本人（利用者）確認要求の受信
(3) 【RA】本人性の確認完了通知の送信
(4) 【端末】本人性確認通知の受信
(5) 【端末】Identity 鍵の生成
(6) 【端末】CA 公開鍵の取得
(7) 【端末】Identity 鍵の証明書発行要求の作成
(8) 【端末】証明書発行要求の送信
(9) 【RA】証明書発行要求の受信
(10) 【RA】証明書発行要求の送信
(11) 【CA】証明書発行要求の受信
(12) 【CA】証明書発行要求の CA 秘密鍵での復号と EK 証明書の取出し
(13) 【CA】EK 証明書の検証
(14) 【CA】Identity 鍵証明書の作成
(15) 【CA】Identity 鍵証明書の暗号化
(16) 【CA】暗号化した Identity 鍵証明書の送信
(17) 【RA】暗号化した Identity 鍵証明書の受信
(18) 【RA】暗号化した Identity 鍵証明書の送信
(19) 【端末】暗号化した Identity 鍵証明書の復号（Identity 鍵のアクティベート）

4. おわりに

本稿では，企業の情報システムの安全性を確保する要素技術の一つとして，TPM による端末認証に着目し，Linux の参照実装やサンプルコードでは考慮されていなかった，地理的に分散している環境でも Identity 鍵証明書を発行するための登録局を含めたプライバシーCA を構築するための設計を行った.

3.2 節で示した手順は，(1)~(4)，(5)~(9)，(10)~(17)，(18)~(19)

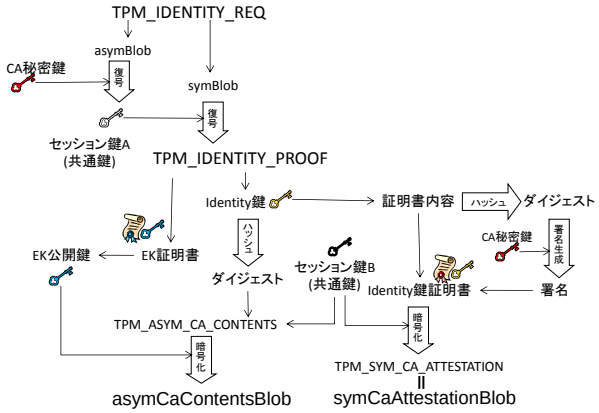

図 2 Identity 鍵証明書発行要求から暗号化した Identity 鍵証明書作成までの処理

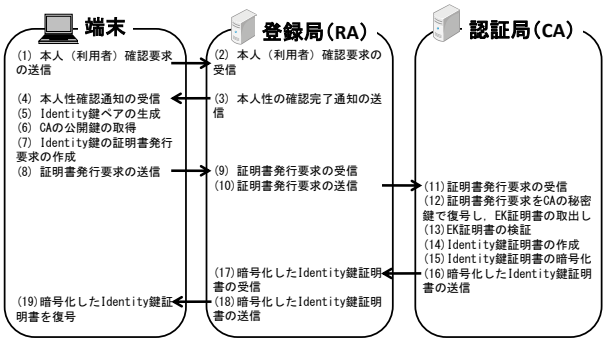

図 3 Identity 鍵証明書発行の手順

と処理が分かれており，独立したサブシステムとして運用が可能となった.

TPM v1.2 に準拠した Infineon 製のチップが搭載されている端末で，IAIK jTSS 0.6 [6]を用いて Java 言語を主に用いて実装している.

参考文献

[1] 中村智久，東川淳紀，“PC 搭載セキュリティチップ(TPM)の概要と最新動向”，IPSJ Magazine Vol.47 No.5, 2006 年 5 月
[2] Privacy CA, http://www.privacyca.com/ (2011/1/8 参照)
[3] JUAS，“第 16 回企業 IT 動向調査 2010（09 年度調査）”，http://www.juas.or.jp/servey/it10/press-pp100409.pdf (2011/1/8 参照)
[4] 情報処理推進機構，PKI 関連技術解説，http://www.ipa.go.jp/security/pki/index.html ，(2011/1/8 参照)
[5] Trusted Computing Group：TPM Main Specification Level 2 Version 1.2, Revision 103, http://www.trustedcomputinggroup.org/resources/tpm_main_specification, (2011/1/11 参照)
[6] Trusted Computing for the Java™ Platform, http://trustedjava.sourceforge.net/index.php?item=jtss/about ，(2011/1/8 参照)