第11部

公開鍵証明書を用いた利用者認証技術

木村 泰司

第１章　moCA WG 2017年の活動

moCA WGはCA(Certification Authority)の振る舞いや証明書の扱いに注目し、WIDEプロジェクトでCAの運用実験を行っているWGである。

moCA WGで運用されているCAのmoCAでは、4種類のクライアント証明書が発行されている。WIDEメンバに発行されるWIDEメンバ証明書、WIDEメンバの秘書さんに発行される秘書さん証明書、一時的にWIDE合宿等に参加するゲスト向けのテンポラリ証明書、WIDE合宿の事務局業務を行うためのWIDE事務局証明書である。サーバ証明書はWIDEサーバ証明書の1種類である。

発行された証明書は、WIDE研究会やWIDE合宿の申し込みなどのユーザ認証やS/MIMEを使った電子メールで使われており、WIDEサーバ証明書はSSL/TLSを使うWebサーバなどで使われている。WIDEプロジェクトで使われているサーバの中には無料のサーバ証明書を利用できる Let's Encrypt が利用されているものがあり、WIDEメンバの間ではWIDEサーバ証明書と使い分けがなされている。

2017年は、定常的になっているmoCAの取り組みに関して、国立情報学研究所の取り組みを参考に見直しを図ると共にWebのPKIにおける信頼に関する議論が12月の研究会で行われた。

第２章　moCAによる証明書発行の概況

WIDEメンバ証明書とWIDEサーバ証明書は1年おきに一斉に発行されている。執筆現在、WIDEメンバ総数は877名で、発行されたWIDEメンバ証明書は890、再発行数は13である。WIDEサーバ証明書は44のドメイン名に対して発行されている。

第３章　現在のWeb PKIとmoCAの乖離 - 12月研究会での話題

moCAは1990年代にWIDEにおけるCAの実験として運用が開始されWIDEメンバに対する証明書発行が行われた[104]。それ以降、WIDEでは認証の定義や発行手続きの重要性に着目した議論が行われてきた。実験としては、S/MIMEへの対応やCA系列の変更、鍵を変更しない証明書の更新といった事も行われた。2006年頃、WIDE共通パスワードの代わりにWIDEメンバ証明書を使ったユーザ認証が本格的に利用されるようになり、moCAの運用は定常的なものとなってきた。

一方、2000年代に入ると、Webブラウザで使われているPKIの信頼性の構造が、認証局監査や監査基準に基づいたものとなり、それに加えて、不正にサーバ証明書が発行された事件を受けて導入された対策技術が普及してきた。近年のWebにおけるPKIと、サーバ証明書が商用サービスとして広く認知される前から運用されてきたmoCAにおけるサーバ証明書には乖離があることが分かっている。2018年12月WIDE研究会で議論された4点を以下にまとめる。

(1) Webブラウザにおけるトラスト・モデルの変化

商用のWebサービスのためのサーバ証明書を発行する認証業務の、要件やガイドラインを定めたWebTrust for CAは、Webブラウザにおけるトラスト

リスト (いわゆる"信頼されたルート証明機関") にCAが登録される条件となった。更にBasic Rquirementといった要件が加わると共に、近年はWebブラウザの開発元による検査も行われるようになった。その結果、運用状態が不明瞭なCAはトラストリストに登録されなくなったものの、その登録の仕組みから、ユーザの知らないCAや一度も使われることのないCAが多数登録された状態になっている。一方でmoCAはユーザ自身が利用するCAを技術的に確認してからWIDE ROOT CAをトラストリストに加える方法を取ってきた。

WIDEで運用されるCAはWebTrust for CAのガイドラインに則ったようなCAではないため、ユーザにとってリスクであるという指摘がある一方で、トラストリストに登録されたCA の下位CAから"*.google.com"や"*.fbi.gov"といったドメイン名を持つ不正なサーバ証明書が発行される事件、すなわちトラストリストを使っていれば安全であるとは言えないような事件が複数起こってきた。攻撃者の視点では、トラストリストに登録されているCAの中で攻撃しやすいCAに偽のサーバ証明書を発行させることができれば、偽のサーバ証明書を有効であるように見せることができるユーザを多く確保できるということになる。ユーザが知らないようなCA を自動的に"トラスト"しておく構造が果たして安全なのかどうかは再検討の余地があると言える。

(2) 失効検証

1990年代には有効期間中に効力をなくした情報をCAが伝える方法としては証明書失効リスト (CRL) が主流であった。その後、オンラインで失効しているかどうかを検証するプロトコルOCSP等が開発され、現在はOCSPやその逐次問い合わせる方式を改良したOCSP Staplingが商用のサーバ証明書においては普及していると言える。

Webブラウザにおける失効検証の実装は未だ統一されてはいないものの、クライアント証明書のように逐次有効性を確認する証明書を扱っているmoCAには改善の余地がある。

(3)サーバ証明書の証明内容

Webのためのサーバ証明書は、基本的に、サーバを識別する文字列とサーバの公開鍵の組み合わせが正しいかどうかを、電子署名の技術を使って確認できる仕組みになっている。Webブラウザにはサーバの識別名としてFQDNが表示されるため、ユーザ自身が意図したサーバにHTTPSでアクセスしているかどうかが確認できる。しかしFQDNの用途は多様化しており、商品名やイベントの名称だけでなく、サーバの役割や組織の略称などを表すようになっている。するとWebブラウザが表示しているFQDNが、ユーザが接続しようとしているサーバなのかそうではないのかが、ドメイン名に関する知識を持たないユーザには区別がつきにくくなっている。

認証局監査の基準及びガイドラインのWebTrust for EVにより知られることになったEV (Extende Validation) 証明書は、FQDNだけでなくそのFQDNを割り当てられている組織の存在が、書類や連絡を通じて確認されているサーバ証明書である。オンラインによる組織の確認が行われているOV (Organization Validation) と、FQDNが確認されているDV (Domain Validation) とは区別される。Google ChromeやFirefoxといったWebブラウザは、EVで確認された組織名をアドレスバーに表示するようになっており、スマートフォンのWebブラウザはEVの組織名のみを表示し、URLを表示しないものもある。エンドユーザの観点ではURLのような確からしさが分かりにくいものよりも、確認されている組織名が表示されている方が分かりやすくてセキュアであると見ることができる。つまりWebサーバ証明書の役割が変わってきていると言える。

一方、DV証明書であればLet's Encryptのような無料の仕組みを利用すると取得でき、自動的に更新されるため、運用に必要な操作が少ない。moCAは現在EVではなく、Let's Encryptを利用しないようなサーバに証明書を発行している。今後も使い分けが進むと考えられる。

(4)不正発行対策

CAにおける不正証明書の発行事件を受けて、Webブラウザの開発元ではトラストリストに入るCAに対する検査や独自の無効化処置を行うようになってきている。またCT (Certificate Transparency) を使って、不正な証明書を検出する動きがある。(1) で述べた通り、トラストリストに入っているCAを信頼の前提にしていいのかどうかは、まだ議論の余地があると言えるが、Webブラウザで警告が表示されるような証明書をmoCAで発行していてもエンドユーザには分かりにくい状況を作ってしまう恐れがある。

Webブラウザにおける不正に発行されたサーバ証明書への対策の動向を踏まえて、moCAでも対策を取っていく必要がある。

第４章　PKIにおける今後の課題 - 12月研究会での議論

前節の4つの乖離は、moCAで発行されるサーバ証明書においてキャッチアップできる項目であると言える。ただし一般的になったサーバ証明書の運用に近付ける位置づけにあり、研究的な技術の進歩に寄与するとは位置づけにくい。またこれまでに起きた事件に対する改善の集合であり、根本的な課題に取り組むものとも言い難い。

これまでmoCA WGで議論されてきて、かつ一般化したWebサーバの証明書においても解決されていないような課題について、12月の研究会で議論されたポイントを以下に述べる。

(1) 認証の対象

Webサーバとパソコンで使われているWebブラウザの間の相互認証が用途として主流であったものから、オンラインの様々な機器の認証といった具合に変わって来ている。IPで通信するノードとしては、必ずしもドメイン名をつけて使うものばかりではなくなってきている。

(2) 本人性

クライアント証明書の発行対象についてはまだ議論が少ない。JPNICにおける認証業務を鑑みても、自然人に対して発行された電子証明書は意外に用途が少なく、むしろ所属や社会的な属性を証明する証明書の方が用途が挙がりやすい。例えば社員証や取引のある会社の社員である事を示すクライアント証明書など。この場合、証明書発行のために物理的に顔を確認する事には意味がない。またサーバ証明書についても、ユーザが視認するものとしてドメイン名ではなく、EV証明書で示される運営組織の名称が重要であるならば、サーバ証明書の証明内容という観点でニーズに変化があると言える。

(3) 鍵管理

クライアント証明書を使うユーザによる鍵管理のライフサイクルは未だに課題が多い。ICカードやSIMカードのように、鍵が物理的なものに結びついているとユーザによる管理や再発行の考え方の整理はしやすいが、ICカードの利用環境は普及には至っていない。

(4) サーバやもののライフサイクル

オンラインの様々な機器においてもそのライフサイクルと証明書のライフサイクルには時間のスケールとして違いがあり、いかに証明書更新を行なっていくかについても検討課題がある。

(5) トラスト

CAのように、オンラインで認知される組織などについて何を根拠にどのように信頼の仕組みを実現すればいいのかについてはまだ道筋が立っていない。Webブラウザによる審査やCTログのような付加もしくは補強の位置づけにある活動はあっても、そもそもユーザが知らないCAをユーザ環境にいれておく必要があるのかといった議論は行われてきていない。

これらについて今後も議論と取り組みがなされる事が望ましいと考えられる。

第 5 章　WIDE Root CA 03フィンガープリント

WIDEプロジェクトにおける電子証明書のトラストアンカーを提供するために運用されている認証局の証明書「WIDE Root CA 03」のフィンガープリントを以下に示す。

SHA-256フィンガープリント
3B:CB:EC:C3:6C:96:ED:D5:A2:98:81:19:C4:C6:F0:4B:DE:AB:43:63:48:D3:7B:05:F9:36:5F:1C:AF:B4:0F:8C

SHA-1フィンガープリント
42:75:7B:24:E3:BB:DB:AB:9E:D7:FE:32:D1:27:18:58:EE:3E:81:66