RSA 署名の技術動向

はじめに

暗号技術は Internet 利用技術におけるセキュリティ確保の要として、単に守秘のためのデータ暗号化ばかりではなく認証・否認防止といった電子商取引を支えるセキュリティのすべてに広く利用されています。なかでも代表的公開鍵暗号方式のひとつである RSA アルゴリズムは、その数式の簡潔さ、背景となる素因数分解問題に関する長年の研究に支えられた信頼感、暗号（署名検証）と復号（署名）のアルゴリズムそしてそれに必要な鍵が対称的でその利用実装が容易である事などから過去 20 年にわたり、文字通り暗号技術のデフォルトスタンダードとして利用されています。

しかしその一方、デフォルトとして多くの利用技術・実装方法が開発・提案されてきたなかで数々の攻撃方法が試みられそれらに対処するための新しい対策が講じられ続けていることは意外に知られていません。Internet 利用の拡大とともに RSA アルゴリズムが採用される機会はますます増加すると思われますがその際、現在進行している技術開発・標準化の動向を正しく把握し正しい実装・利用を行わなければ単にインターオペラビリティ上の問題だけではなく不適切な実装がセキュリティ上の問題の原因ともなりかねません。

RSA アルゴリズムは特許の切れた成熟しきった技術ではなく常に刷新される新たなセキュリティ技術のコアであることを知っていただくために、近年 RSA アルゴリズムの特に署名アルゴリズムに関しての技術動向をご紹介したいと考えます。

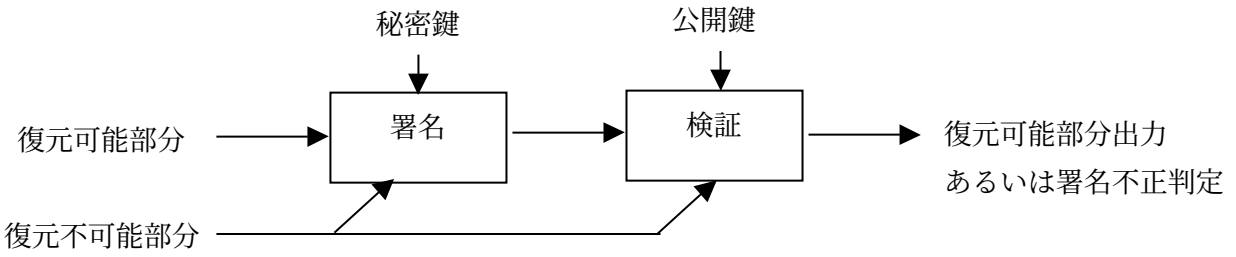
１．デジタル署名の基本モデル

１－１．デジタル署名の基本要素

公開鍵暗号にもとづくデジタル署名の仕組みは大きく 3 つの段階に分けることができます。

- 公開鍵/秘密鍵生成
- 秘密鍵による署名
- 公開鍵による（署名）検証

また、デジタル署名は、署名の対象によって署名添付型、メッセージ完全復元型、メッセージ部分復元型の 3 つに分類されます。署名添付型は送付メッセージと別途生成された署名をいっしょに送付する方式です。メッセージ完全復元型は送付された署名よりメッセージを復元可能な方式、部分復元型はメッセージの一部は署名より復元し他の一部はメッセージとして送付される方法です。



デジタル署名の多くはトラップドア付一方向関数を用います。一方向関数とは関数の計算は容易だが逆関数の計算が困難であるような関数を言います。トラップドア付一方向関数は、ある情報（トラップドア）がある場合には逆関数の演算が簡単になるような一方向関数です。RSA では $f(x)=x^e \bmod n$（n=pq、ｐとｑは素数、e は(p-1)(q-1)と互いに素な自然数）が一方向関数として用いられますが、これは $d=e^{-1} \bmod \mathrm{lcm}(p-1,q-1)$とすると、$f^{-1}(x)=x^d \bmod n$ によりトラップドアを構成できます。

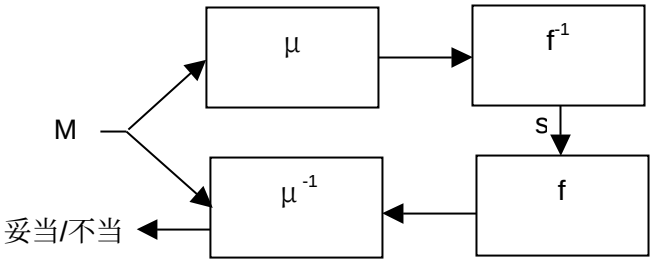
１－２．署名演算

添付型署名では、伝送メッセージより“ メッセージの代表 ”となる値を生成し、それにトラップドア演算（逆関数計算）を施して署名とします。すなわち、$s=f^{-1}(\mu(M))$です。
ここでエンベッド演算 μ(M)はメッセージから“ メッセージの代表 ”を計算する演算です。簡単な例ではメッセージをハッシュしてパディングしたものなどが用いられます。本稿でご紹介するいくつかの現行 RSA 署名方法は主にそれぞれの μ の計算に違い・特徴があると言えます。たとえば μ を乱数化するなどしてセキュリティを高めることなどが行われます。
復元型では、$s=f^{-1}(\mu(M_r,M_{nr}))$　（M_r は復元可能部分、M_{nr} は不可能部分）となります。

１－３．署名検証

署名の検証は、添付された署名を一方向関数に入力し、その結果が正当であるかどうかをチェックするものです。添付型では、f(s)を計算し、M に対する妥当性μ^{-1}（単純な例ではμ(M)を再計算して比較）をチェックして署名の妥当・不当を判定します。復元型では M_{nr} に対する妥当性をチェックすると同時に正しく M_r が復元できるかを検証します。

添付型の流れ例



１－３．エンベッド演算の性質

エンベッド演算にはハッシュ関数と同様の性質、すなわち一方向性と衝突困難性が必要です。また当然の事ながらトラップドア関数とうまく適合する必要があり、理想的には乱数性を持つことが求められます（後述）。署名の方法によってはアルゴリズムを識別する手段を実装する場合がありますが、不注意なメカニズムの実装がセキュリティ上の弱点となった例があり注意が必要です。

２．RSA 署名の安全性

２－１．RSA 署名の乗法性

RSA の関数 $f(x)=x^e \bmod n$ は分配法則を満たします。すなわち、$f(xy \bmod n)=f(x)f(y) \bmod n$、$f^{-1}(xy \bmod n)=f^{-1}(x)f^{-1}(y) \bmod n$ が成り立ちます。この性質を利用した多くの署名偽造攻撃が研究されて参りました。反対にそれらの研究を通して RSA 署名を改良し安全性の向上を高めるための方策が絶え間無く行われています。

２－２．署名偽造

デジタル署名に対する攻撃は一般的に偽造と呼ばれます。偽造とは署名者の秘密鍵無しに署名を生成することで、任意のメッセージに対し署名を生成する一般的な偽造と、特定のメッセージに対して署名が生成できる存在的偽造に分けられます。一般的偽造は論外にしても存在的偽造が不可能であることがデジタル署名方式に求められる安全性です。また、偽造にいたる攻撃の方法も幾つかに分類されますが、中でも攻撃者が選んだメッセージに対しての署名を署名者から入手できる環境で、それらの署名をもとに署名の偽造を行う“選択文書攻撃”がもっとも強力な攻撃だといわれており、選択文書攻撃による存在的偽造が不可能であるデジタル署名方式が求められます。

２－３．RSA 署名に対する攻撃

（１）小素数法

RSA 署名の乗法性によりμ(M)がμ(Mi)に素因数分解できる場合、署名も$\Pi \mu(M_i)^d \bmod n$となり選択的文書攻撃により入手した多数の署名を利用して Mi に対応する署名を得ることができます。この結果攻撃者は Mi の署名を組み合わせてμが素数の積となる任意のメッセージの署名を偽造可能となります。この攻撃法は早くから知られており、μを適切に選択する（たとえばμの一部としてハッシュを使う、冗長性を持たせる）ことにより防御可能であるため、このままでは現在利用されている署名法に対しての効果は限られています。

（２）より一般化された攻撃法

ISO9796 はμに冗長性を持たせた復元型署名の国際標準でしたが、μのフォーマットに着目しそれを満足しつつ小さな素因数をもつデータを集めてその署名を集めることにより小素数法と同様の方法が適用可能となるという発表がなされました。ISO9796 はこれによりその安全性に疑問がもたれ廃止となっています。

ISO9796-2 はμの一部にハッシュを用いる復元型署名の標準ですが、μの素因数分解ではなくそれにある変形を加えた式が素因数を持つように M を選択しその署名を集めた後、小素数法と同様の方法を適用する攻撃が提案されています。

（３）整数関係法

μとｆを含むある方程式が解ける場合に有効な攻撃で、ISO9796-1 に対し大変少ない選択文書で偽造可能となる例が発見されました。

２－４．可能安全性の数学的証明

（１）証明可能安全性

ISO9796 等はいずれも特定の攻撃や解析手法への対応がなされたものとして提案されましたが、上述のようにその後の研究で新たな（素因数分解よりも）効果的な攻撃法が開発されています。その他でも署名や暗号に対する攻撃方法が研究され、それらの脅威が確認されるにつれて、これまでのような経験に基づいた署名アルゴリズムの検討ではなく、数学的にセキュリティの根拠が確認できる方式の必要性が認識されるようになりました。

この数学的証明としては、f^{-1} の計算を署名偽造の方法に“ 変換 ”する、すなわち、偽造アルゴリズム F を与えて逆関数計算アルゴリズム I を構築できるということを示すことが一般的に行われています。その逆として、逆変換が困難であることから偽造困難性を導くことで証明可能安全性を示すことが可能となります。

（３）ランダム・オラクルモデル

現在提案されている証明可能安全性のほとんどは、逆変換の困難性のほかに、エンベッド演算を行う関数（例えばハッシュ関数）が（真性の）乱数を発生させることを仮定として用いています。同一入力に対し同一の乱数を出力する関数をランダム・オラクルを呼びます。ランダム・オラクルモデルにより、証明可能となる偽造アルゴリズムを一般化するこ

とが可能です。偽造アルゴリズムからはランダム・オラクルはブラックボックスとして扱われその中を覗くことができません。つまりハッシュ関数の入出力に相関が無いため逆関数の計算に手がかりを与えなくなります（値の隠蔽）。
ランダム・オラクルの仮定は仮想的であり現実にはそのような数学関数は知られておりません。実際に利用されることの多いハッシュ関数を前提に証明可能安全性を有するアルゴリズムを構築する研究が進められています。
（４）RSA 暗号の一方向性（逆変換の困難性）
RSA 関数の逆関数を計算することが計算量的に困難であるという仮定も数学的に証明されてはいませんが RSA 署名の安全性（偽造の困難性）が RSA 関数解読の困難性に還元できるということは数学的妥当性・厳密性は別にして心やすまる論理です。

３．現在の署名方法

現在多数の RSA アルゴリズムに基づく署名アルゴリズムは考案され規格化されていますが、その中で署名添付型を中心に主なものを紹介したいと考えます。

３－１．基本 RSA 署名

μ(M)としてハッシュ関数 Hash(M)だけを利用するコアの RSA アルゴリズムによる署名の仕組みについてはご存知の方も多いと思いますが、これは教科書的説明の目的以外に利用されるべきではありません。典型的な Hash 関数のサイズでは先の小素数法等に対して脆弱性がありセキュリティ上大変問題があります。

３－２．ANSI X9.31

μ(M)=6b bb … bb ba ‖ Hash(M) ‖ 3x cc、SHA-1 に対しては x=3、RIPEMD では x=1 となります。アドホックなデザインです。乗法性に起因する偽造に対しては耐性があります。IEEE1363、ISO/IEC14888-3、NIST FIPS186-1 等多くの標準にも採用されています。“強い素数”を要求していることに関しては反対意見があります。

３－３．PKCS #1 v1.5

PKCS は RSA セキュリティが中心となりまとめた暗号技術標準の総称です。#1 から#15(途中欠番あり)まであり、暗号アルゴリズムからデジタル署名の実装、鍵の保存法、暗号デバイスとのインターフェイス等幅広く規定しています。私企業が中心となってまとめた規格ながら多くの標準のもととなりまた数々のセキュリティ製品の開発に参照されているものです。その中で#1 が RSA アルゴリズムの暗号化・署名の手順を定めています。
μ(M)=00 01 ff… ff 00 ‖ HashAlgID ‖ Hash(M)、ANSI 同様アドホックなデザインです。乗法性に起因する偽造に対して耐性があります。SSL や S/MIME の署名方式として規定されている他 IEEE1363a にも含まれています。PKCS #1 v2.0 でもサポートされています。

ANSI9.31、PKCS #1 v1.5 ともにμは決定的でハッシュ ID を含んでいます。両者ともアドホックなデザインで乗法性を利用した偽造に耐性があるなど共通点が多くあります。

３－４．Bellare-Rogaway FDH(Full Domain Hashing)

μ(M)=Full-Length-Hash(m)。ハッシュのサイズが法ｎと同一です。証明可能安全性に基づきデザインされています。ハッシュはランダム・オラクルの仮定に従います。変形版が IEEE1363a、PKCS #1 v2.1 ドラフトに含まれています。実際の Hash 関数として既存 Hash の連結等が提案されています。

３－５．Bellare-Rogaway PSS

μ(M)≈H‖G(H)　salt、H=Hash(salt,M)、salt は乱数、G はマスク生成関数。証明可能安全。変形版が IEEE1363a、PKCS #1 v2.1 ドラフトに含まれています。

FDH と PSS はともに証明可能安全性に基づきます。FDH のμは決定的ですが PSS は確率的です。PSS の方がより強い安全性証明を有し Hash のセキュリティへの依存度は低いようです。PSS の変形である PSS-R は復元型（完全、部分）をサポートします。

1998 年、PKCS #1 v1.5 を暗号化に利用したある特定の実装形態を想定した場合（復号結果の先頭 2 バイトをデータの妥当性チェックに利用しエラー対応を行う処理が施されしかも不正データの繰り返し伝送への対処が十分ではない場合）に、セキュリティ強度が低下する可能性があることが発表されました。これによる実際の被害は報告されていませんが RSA セキュリティはこれを潜在的なセキュリティに対する脅威と認識して、これへの対処として証明可能安全性に基づく暗号化処理である RSA－OAEP を採用して PKCS #1 v2.0 に盛り込みました。その後 OAEP の証明可能安全性の数学的証明に小さな誤りが見つかるなどしておりますが、現在までのところ証明可能安全性は覆されていません。PKCS #1 v2.0 では暗号化部分が先に証明可能安全性を取り入れており、署名に関しては v2.1 ドラフトに盛り込まれている状態です。

３－６．IEEE P1363a 版 PSS

μ(M)= G(H)　[00 … 01‖salt]‖H‖bc、H≈Hash(salt,Hash(M))、salt は乱数、G はマスク生成関数。Salt は M ではなく Hash(M)と結合されています。これはセキュリティ上の理由（フォールト解析攻撃への耐性）と実装上の理由（ワンパス処理）によるものです。この Hash 部の内容により証明可能安全性かどうかが変わります。

３－７．RSA 復元型署名

基本の RSA アルゴリズムを用いた復元型署名は添付型と同様勧められません。ISO9796-x については上に見てきた通りです。9796-2 は十分大きな Hash 値に対しては乗法性に基づく偽造に対しても十分なセキュリティを持っていると考えられます。PSS の復元型版として PSS-R が IEEE1363a や ISO9796-2 改定ドラフトに盛り込まれています。

４．RSA 署名の標準化動向

PSS が証明可能安全性を有する方式として優れていることは先に述べてきた通りですが、現実には ANSI9.31 が多くの標準として採用され PKCS #1 v1.5 が SSL のプロトコルとして広範囲に利用されています。この現状からどのように理論的に優れた PSS の利用に推移して行くかは大きな課題であり、その解決には標準化に関わっている多くの団体間の調整や膨大な利用者環境の移行が必要で、長い時間を要することは明らかです。

ANSI9.31 や PKCS #1 v1.5 は良く考えられてデザインではありますが、将来の万が一の危険性を考えると、短期的にはインターオペラビリティ確保の観点からそれらの利用を続けるにしても、明確な長期的(例えば 2-3 年)移行計画を策定し PSS 等へ移行することが必要です。

PSS、PSS-R の標準化自体は、IEEE1363a、PKCS #1 v2.1、ISO9796-2 改定等により進んでおり ANSI や FIPS、IETF での採用も計画されています。また今後、AES の採用や新 Hash の開発など、アップグレードのタイミングを見るのに都合の良い機会も少なくありません。これからの RSA 署名標準化の動きを注意深く見守って行く必要があります。