ISO/IEC 9796-2 (Scheme 1) 署名の偽造について (その１)

杜撰な研究者

2009 年 11 月 29 日 (日)

ISO/IEC 9796-2 (Scheme 1) 署名の偽造法 (CNTW 偽造法) を理解できたような気がするため、備忘録代わりに要点をメモしておきます。しかし CNTW 偽造法については産総研や富士通から詳細な分析報告がなされているので、本ブログではこの分析報告に出てこない補助情報を書き残しておきたいと思います　どう考えても一回では終わらなそうなので、まず今回は CNTW 偽造法のベースとなった Desmedt-Odlyzko による偽造法 (DO 偽造法) から。

まずは RSA 署名の復習から。p, q を $k/2$ ビットの素数、$N = p \times q$ を k ビットの合成数として、公開鍵を (N, e)、秘密鍵を (d, p, q) とおきます。このとき、メッセージ m に対する RSA 署名は $s = u(m)^d \bmod N$ によって生成されます。また、メッセージ m に対する署名 s の検証は s^e と $u(m)$ が $\bmod N$ で等しいかをチェックします。ここで $u(m)$ はパディングと呼ばれる関数で、暗号の教科書では $u(m) = m$ として説明されることが多いのですが、安全性に問題があるため、現実にはもう少し複雑な関数が使用されます。

さて、本題の Desmedt-Odlyzko による偽造法 (DO 偽造法) ですが、攻撃の最終目標は偽造署名を算出することです。前提として、攻撃者はメッセージを選択できることと、署名オラクルを利用できることを仮定します。具体的には、攻撃者は最終的に

$$s^* = D \times s_1^{e_1} \times s_2^{e_2} \times \cdots \times s_L^{e_L} \bmod N$$

という関係式から偽造署名 s^* を算出するので、攻撃者は係数 D と指数 $e_1, e_2, \ldots, e_L$ を求めれば十分です。ここで $s_1, s_2, \ldots, s_L$ はメッセージ $m_1, m_2, \ldots, m_L$ に対応する (同じ秘密鍵のもとでの) 署名です。係数 D と指数 $e_1, e_2, \ldots, e_L$ を求めるには、メッセージに関する関係式

$$u(m^*) = D^e \times u(m_1)^{e_1} \times u(m_2)^{e_2} \times \cdots \times u(m_L)^{e_L} \bmod N$$

を利用します。つまり DO 攻撃の目標は、この関係式を満たす係数 D と指数 $e_1, e_2, \ldots, e_L$ を求めることです。

上の関係式を求めるために、DO 偽造法は $u(m)$ の素因数分解を利用します。その方法を説明する前に、素因数分解関係の記号を定義しておきましょう。L 個の素数を小さい順にならべて得られる集合を Prime_L とかき、その要素を $p_1 = 2, p_2 = 3, \ldots, p_L = B$ とおきます。ある自然数が B 以下の素数のべき積として素因数分解できるとき、その自然数は B-smooth であると言います。例えば 15 は 3 と 5 の積として素因数分解できますので、15 は 5-smooth です。

DO 偽造法は、$u(m)$ が B-smooth となるようなメッセージ m を $L+1$ 個以上収集します。得られたメッ

セージとその素因数分解を

$$\begin{array}{rcccccccc}
u(m_1) & = & p_1^{v(1,1)} & \times & p_2^{v(1,2)} & \times & \ldots & \times & p_L^{v(1,L)} \\
u(m_2) & = & p_1^{v(2,1)} & \times & p_2^{v(2,2)} & \times & \ldots & \times & p_L^{v(2,L)} \\
\ldots & & & & & & & & \\
u(m_L) & = & p_1^{v(L,1)} & \times & p_2^{v(L,2)} & \times & \ldots & \times & p_L^{v(L,L)} \\
u(m_{L+1}) & = & p_1^{v(L+1,1)} & \times & p_2^{v(L+1,2)} & \times & \ldots & \times & p_L^{v(L+1,L)}
\end{array}$$

と表し、$u(m_i)$ の素因数分解に登場する指数から得られる L 次元ベクトル

$$V_i = (v(i,1), v(i,2), ..., v(i,L))$$

を考えます。このベクトルは $L+1$ 個ありますので、高い確率で線形従属となり、あるベクトル V_t を他のベクトルの線形結合として

$$V_t = e \times V_0 + b_1 \times V_1 + b_2 \times V_2 + \cdots + b_L \times V_L$$

と表すことができます。ここで $V_0 = (v(0,1), v(0,2), \ldots, v(0,L))$ は定数ベクトル、$b_1, b_2, \ldots, b_L$ は e 未満の整数係数となります (線形結合の計算手順の説明はここでは省略しますが、以下の数値例から理解できると思いますが)。

このとき

$$\begin{array}{rcccccccccc}
D & = & & & p_1^{v(0,1)} & \times & p_2^{v(0,2)} & \times & ... & \times & p_L^{v(0,L)} \\
u(m_t) & = & D^e & \times & u(m_1)^{b_1} & \times & u(m_2)^{b_2} & \times & \ldots & \times & u(m_L)^{b_L}
\end{array}$$

とおくことで、偽造署名

$$s_t^* = D \times s_1^{b_1} \times s_2^{b_2} \times \cdots \times s_L^{b_L} \bmod N$$

が得られます。

うーん、数式だらけで全然わかりませんね　とりあえず押さえておきたいのは、DO 偽造法はメッセージ $u(m)$ が素因数分解できることを利用している点です。単純な RSA 署名であれば $u(m) = m$ となっていますので、$u(m)$ が素因数分解できるようなメッセージを収集することは簡単です。しかし $u(m)$ がもっと複雑な場合 ($u(m)$ のビット長が大きな場合)、その素因数分解は困難となり、DO 偽造法も適用できなくなってしまいます。ISO/IEC 9796-2 (Scheme 1) は $u(m)$ が $k-1$ ビットとなるように設計されているので、DO 偽造法は適用できないのです。これが DO 偽造法の特徴でもあり限界でもあります。

最後に DO 偽造法の処理内容を数値例を用いて説明していきます。以下では $u(m) = \mathrm{SHA16}(m)$ (SHA16 は出力長が 16 ビットのハッシュ関数で、具体的には SHA-1 の下位 16 ビットを用います)、$e = 3$、$L = 7$ (つまり $\mathrm{Prime}_L = \{2, 3, 5, 7, 11, 13, 17 = B\}$) とします。初めの処理は、$u(m_i)$ が B-smooth となるようなメッセージ m_i を $L+1$ 個以上収集することです。$m = 1, 2, ...$ と変化させていくと、以下の 8 個のメッセー

ジを得ることができます:

$$
\begin{array}{lllllllllllllllllll}
u(m_1) & = & u(40) & = & 60480 & = & 2^6 & \times & 3^3 & \times & 5 & \times & 7 & & & & & & \\
u(m_2) & = & u(64) & = & 9464 & = & 2^3 & & & & & \times & 7 & & & \times & 13^2 & & \\
u(m_3) & = & u(124) & = & 10200 & = & 2^3 & \times & 3 & \times & 5^2 & & & & & & & \times & 17 \\
u(m_4) & = & u(159) & = & 22253 & = & & & & & & \times & 7 & \times & 11 & & & \times & 17^2 \\
u(m_5) & = & u(163) & = & 726 & = & 2 & \times & 3 & & & & & \times & 11^2 & & & & \\
u(m_6) & = & u(183) & = & 11900 & = & 2^2 & & & \times & 5^2 & \times & 7 & & & & & \times & 17 \\
u(m_7) & = & u(257) & = & 54054 & = & 2 & \times & 3^3 & & & \times & 7 & \times & 11 & \times & 13 & & \\
u(m_8) & = & u(264) & = & 12716 & = & 2^2 & & & & & & & \times & 11 & & & \times & 17^2
\end{array}
$$

次に、これら 8 個の素因数分解から線形結合を求めるために、次のような行列を考えます。この行列の各行は上の素因数分解の各行の指数を並べたものです。また右側には、どの行とどの行を足し合わせたかを記録するために、単位行列を並べておきます。

$$
\left(\begin{array}{ccccccc|cccccccc}
6 & 3 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
3 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
3 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
2 & 0 & 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 3 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
2 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right)
$$

次に各成分を mod$e$ します。

$$
\left(\begin{array}{ccccccc|cccccccc}
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
2 & 0 & 2 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
2 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right)
$$

以下、Gauss 消去によって線形結合を求めていきます。まず、5 行目をピポットにして 1 列目を消去します。

$$\left(\begin{array}{cccccccc|cccccccc}
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\mathbf{1} & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 2 & 1 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 2 & 0 & 1 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1
\end{array}\right)$$

次に、3 行目をピポットにして 2 列目を消去します。

$$\left(\begin{array}{cccccccc|cccccccc}
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \mathbf{1} & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 2 & 1 & 2 & 1 & 1 & 0 & 0 & 1 & 0 & 2 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 1
\end{array}\right)$$

次に、1 行目をピポットにして 3 列目を消去します。

$$\left(\begin{array}{cccccccc|cccccccc}
0 & 0 & \mathbf{1} & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 2 & 2 & 1 & 1 & 1 & 0 & 1 & 0 & 2 & 0 & 1 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 & 1 & 2 & 0 & 2 & 0 & 1 & 0 & 0 & 1
\end{array}\right)$$

次に、2 行目をピポットにして 4 列目を消去します。

$$\left(\begin{array}{cccccccc|cccccccc}
0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \mathbf{1} & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 2 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 2 & 2 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 2 & 0 & 1 & 1 & 1 & 1 & 0 & 2 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 1 & 2 & 1 & 2 & 0 & 1 & 0 & 0 & 1
\end{array}\right)$$

次に、4 行目をピポットにして 5 列目を消去します。

$$\left(\begin{array}{ccccccc|cccccccc} 0&0&1&1&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&1&0&2&0&0&1&0&0&0&0&0&0\\ 0&1&2&0&0&0&1&0&0&1&0&0&0&0&0\\ 0&0&0&0&\mathbf{1}&1&2&0&2&0&1&0&0&0&0\\ 1&1&0&0&2&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&2&2&0&1&2&1&1&1&0&0\\ 0&0&0&0&0&1&0&1&0&1&1&2&0&1&0\\ 0&0&0&0&0&2&1&2&1&2&0&1&0&0&1 \end{array}\right)$$

次に、7 行目をピポットにして 6 列目を消去します。

$$\left(\begin{array}{ccccccc|cccccccc} 0&0&1&1&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&1&0&2&0&0&1&0&0&0&0&0&0\\ 0&1&2&0&0&0&1&0&0&1&0&0&0&0&0\\ 0&0&0&0&1&1&2&0&2&0&1&0&0&0&0\\ 1&1&0&0&2&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&2&1&1&0&2&0&1&1&0\\ 0&0&0&0&0&\mathbf{1}&0&1&0&1&1&2&0&1&0\\ 0&0&0&0&0&0&1&0&1&0&1&0&0&1&1 \end{array}\right)$$

最後に、8 行目をピポットにして 7 列目を消去します。

$$\left(\begin{array}{ccccccc|cccccccc} 0&0&1&1&0&0&0&1&0&0&0&0&0&0&0\\ 0&0&0&1&0&2&0&0&1&0&0&0&0&0&0\\ 0&1&2&0&0&0&1&0&0&1&0&0&0&0&0\\ 0&0&0&0&1&1&2&0&2&0&1&0&0&0&0\\ 1&1&0&0&2&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&1&2&0&0&0&1&2&1\\ 0&0&0&0&0&1&0&1&0&1&1&2&0&1&0\\ 0&0&0&0&0&0&\mathbf{1}&0&1&0&1&0&0&1&1 \end{array}\right)$$

このとき、6 行目の成分が全て 0 になっていますので、線形結合

$$0 = 1 \times V_1 + 2 \times V_2 + 0 \times V_3 + 0 \times V_4 + 0 \times V_5 + 1 \times V_6 + 2 \times V_7 + 1 \times V_8 \bmod e$$

つまり

$$V_6 = -1 \times V_1 - 2 \times V_2 - 2 \times V_7 - 1 \times V_8 \bmod e$$

が得られます。よって、各ベクトルの成分ごとに比較することで、整数上の線形結合

$$V_6 = e \times V_0 + 2 \times V_1 + 1 \times V_2 + 1 \times V_7 + 2 \times V_8$$

が得られます。ここで $V_0 = (-6, -3, 0, -1, -1, -1, -1, -1)$ となります。

以上より、

$$\begin{aligned} D &= 2^{-6} \times 3^{-3} \times 7^{-1} \times 11^{-1} \times 13^{-1} \times 17^{-1} \\ u(m_6) &= D^e \times u(m_1)^2 \times u(m_2)^1 \times u(m_7)^1 \times u(m_8)^2 \end{aligned}$$

という関係式が得られます。ここまでの計算に N や d の値を用いていないことに注意して下さい。

最後は偽造署名の算出です。公開鍵 $N = 281467359833221$ と $s_1 = 992405412287$, $s_2 = 108524445318$, $s_7 = 912311723672$, $s_8 = 362819320988$ が得られたとすると、偽造署名

$$s_6^* = D \times s_1^2 \times s_2^1 \times s_7^1 \times s_8^2 \bmod N = 815606114320$$

が算出できます。実際、$(s_6^*)^e \bmod N = 11900 = u(m_6)$ となり、偽造署名の正当性が検証できます。偽造には秘密鍵を一切用いていないことに注意して下さい。