プライバシ保護技術

土井洋(情報セキュリティ大学院大学)

今回の入門講義では，匿名性を有する電子署名とはどんな技術か，またその仕組みについて解説したい．このような匿名性を有する電子署名を構成する場合には，ゼロ知識証明が広く利用される．具体例として，離散対数問題の困難性を利用するゼロ知識証明を取り上げ，匿名性を有する電子署名の構成方法をみる．

また，署名者集合のサイズに署名長が依存しない方法の構成方法の概要と，それに深く関係する幾つかの数論問題についても述べたい．

講義中に説明が不足していた部分について追記をしました．

電子署名とプライバシ保護

- 電子署名とプライバシ保護
  - 電子署名を用いると，誰が，どのような文書に対して署名を行ったかを，誰でも確認できる．
  - 実世界における署名よりも強力な機能であるが，プライバシ保護の観点からはこの機能を一部制限したい場合もある．
    - 電子商取引において，署名者が作成した電子署名を収集された場合は，署名者の意図に反して，商取引の履歴が第三者に知られるといった不都合が生じるかもしれない．
- 目標
  - 匿名性を有する署名の構築
    - 署名者が誰であるかを検証できない機能の実現

復習(HVZKIP)

(p,q,g,y)に対して$y=g^x \bmod p$となるxを知っている

P(証明者)　　V(検証者)

(秘密の情報x)

$a = g^r \bmod p$ (r:乱数)

a →

← b

$c= r+bx \bmod q$ →

$g^c \stackrel{?}{\equiv} ay^b \pmod p$

Pが離散対数xを知らない場合，受理される確率は1/q

qを2^{160}程度にすれば健全性はOK

復習(知識の署名)

公開鍵簿　Pの公開鍵:y　　Pは$y=g^x \bmod p$となるxを知っている

P(証明者)　　V(検証者)

(秘密の情報x)

$a = g^r$ (r:乱数)

b=h(M,p,q,g,y,a)

c= r+bx mod q

(b,c),M →

①$a=g^c/y^b \bmod p$

②c=h(M,p,q,g,y,a)が成り立つか検証する

乱数生成時，Pがメッセージを(理想的な)ランダム関数へ入力している．

Mの値が変わったら，乱数bも変わることに注意．

Schnorr署名と呼ばれる方法とほぼ同じ

知識の署名(記号について)

- 知識の証明:
  - PK{(α):$y \equiv g^{\alpha} \pmod p$}
    - zero-knowledge Proof of Knowledge of integer α such that $y \equiv g^{\alpha} \pmod p$
- 知識の署名
  - SPK{(α):$y \equiv g^{\alpha} \pmod p$}(m)
    - Singnature of Proof of Knowledge of integer α such that $y \equiv g^{\alpha} \pmod p$

知識の証明の拡張(OR)

- (p,q,g_1,g_2,y_1,y_2)に対して$y_1=g_1^x \bmod p$ または $y_2=g_2^x \bmod p$となるxの知識の証明
  - いずれの知識を持っているかを検証者が識別できないことを目標
    - $y_1=g_1^x \bmod p$となるxの知識
    - $y_2=g_2^x \bmod p$となるxの知識

    } 証明者Pがいずれの知識を持っているか検証者は識別できない
- 戦略
  - 一方の知識(例えば $y_2=g_2^x \bmod p$となるxの知識)を持っているので，こちらのほうは証明を構成可能
  - 他方の知識(例えば $y_1=g_1^x \bmod p$となるxの知識)は持っていないので証明不可能だが，模倣はできる.
  - そこで，「本当の証明」と「模倣」を並べる.

知識の証明の拡張(OR)

(p,q,g_1,g_2,y_1,y_2)に対して$y_1=g_1^x \bmod p$ か$y_2=g_2^x \bmod p$となるxの知識

P(証明者)　　V(検証者)

(秘密の情報x　$(y_2=g_2^x \bmod p)$)

$$\begin{cases} a_1 = g_1^{c_1} y_1^{-b_1} \bmod p \\ a_2 = g_2^{w} \bmod p \end{cases} \xrightarrow{a_1,a_2} \quad 0 \leqq b < q \text{をランダム}$$

$$\xrightarrow{b}$$

$$\begin{cases} b_2 = b - b_1 \bmod q \\ c_2 = w + x b_2 \bmod q \end{cases} \xrightarrow{b_1,b_2,c_1,c_2} \begin{cases} b \stackrel{?}{\equiv} b_1 + b_2 \pmod q \\ g_1^{c_1} \stackrel{?}{\equiv} a_1 y_1^{b_1} \pmod p \\ g_2^{c_2} \stackrel{?}{\equiv} a_2 y_2^{b_2} \pmod p \end{cases}$$

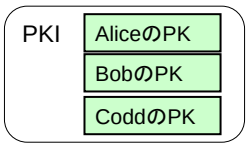
知識の証明の拡張(OR)の証明

- 完全性
  - 証明者に一方の秘密xの知識があれば，プロトコルどおりに振舞えば，検証は成功する.
- 健全性
  - 2種類以上の(異なる)b,b’に対して答えることのできる証明者は秘密xを知っている.
    - b, b’に対して，正しく答えることができたとせよ(今回の場合は，(b_2,b_2',c_2,c_2')をb,b’に対して答えることになる.
    - すると，xは次の式で求まる.

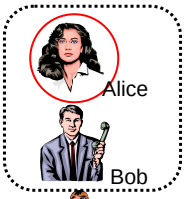
$$x = \frac{c_2 - c_2'}{b_2 - b_2'} \bmod q$$

- ゼロ知識性(special honest-verifier zero knowledge)
  - $b,(b_1,b_2,c_1,c_2),(a_1,a_2)$の順に生成した対話は，本物の対話と分布が同じ.
  - 本物でない方の模倣を行うことができる.

1-out-of-n署名



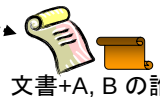
1. 公開鍵登録



集合S={A, B}

- 署名者集合S
  - 署名を生成する可能性がある署名者の集合
- 設計目標
  - 署名者集合Sの誰かが作成した署名であることを検証可能
  - 署名者集合Sの誰が作成したかは検証不可能

2. 署名生成処理



文書+A, B の誰かが作成した署名

検証者

3. A, Bの誰かが署名したことを検証

1-out-of-n証明(知識の証明(n=2))

(p,q,g,y_1,y_2)に対して$y_1=g^x \bmod p$ か$y_2=g^x \bmod p$となるxの知識

P(証明者)

(秘密の情報x $(y_2=g^x \bmod p)$)

V(検証者)

$$\begin{cases} a_1 = g^{c_1} y_1^{-b_1} \bmod p \\ a_2 = g^{w} \bmod p \end{cases} \xrightarrow{a_1,a_2}$$

0≦b <qをランダム

$$\xrightarrow{b}$$

$$\begin{cases} b_2 = b - b_1 \bmod q \\ c_2 = w + x b_2 \bmod q \end{cases} \xrightarrow{b_1,b_2,c_1,c_2}$$

$$\begin{cases} b \overset{?}{\equiv} b_1 + b_2 \pmod q \\ g^{c_1} \overset{?}{\equiv} a_1 y_1^{b_1} \pmod p \\ g^{c_2} \overset{?}{\equiv} a_2 y_2^{b_2} \pmod p \end{cases}$$

スライド7と全く同じ

1-out-of-n署名(n=2)

- システムパラメタ
  - p,q,g
- 署名者の公開鍵
  - y_1(Alice)
  - y_2(Bob)
- 署名
  - (b_1,b_2,c_1,c_2)
- 検証

$$b_1 + b_2 \stackrel{?}{\equiv} h(M, p, q, g, g^{c_1} y_1^{-b_1}, g^{c_2} y_2^{-b_2}) \pmod{p}$$

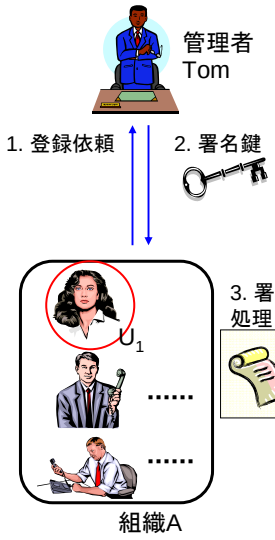
スライド10の知識の証明を非対話化

nを増やすと･･･

- n=2の場合を容易に拡張できる
- 署名サイズはnに比例
  - $(b_1,b_2,\cdots,b_n,c_1,c_2,\cdots,c_n)$
    - $|b_i|=|c_i|=160$
  - 署名者集合のサイズが大きくなると，非現実的

- 改良目標
  - 署名者集合のサイズ(=n)に比例しない

グループ署名の考え方('91～)



1. U_1は管理者Tomに組織Aへの登録を依頼する.
2. Tomは参加OKなら署名鍵SK_1を発行する.
3. U_1は，署名鍵SK_1を使い署名を生成する.
4. 検証者は組織Aの誰かの署名であることがわかる.
5. 不適切な文書への署名者がだれかを，(特殊な管理者Ozzieが)開示できる.

4. 組織Aの誰かが署名したことを検証

提案当時の方式は署名者集合のサイズに依存

目標(informal)

- 管理者Tは署名者集合Sに含まれる各署名者(U_1,U_2,U_3,･･･)の登録依頼を受けて，署名鍵SK_1,SK_2,SK_3,･･を生成できる.
- 管理者Tは検証鍵PK_Gを生成できる
- 署名者U_iは署名鍵SK_iを使い署名σを生成できる
- 検証者は検証鍵PK_G，署名σと文書Mから「署名σは署名者集合Sの誰かが署名鍵を用いて生成したこと」を検証できる.

- 署名σを署名者集合Sに含まれていない人が偽造できない.
- 署名σから署名者を特定できない.

- 特殊な管理者Ozzieのみは，署名σから，署名者が誰かを開示できる.

- 署名長は，署名者集合のサイズに依存しない.

構成例

- 署名者は秘密xを選び，y=F(x)を計算し，yを管理者に送る.
- 管理者はyに対してv=S(y)を計算し，vを署名者に送る.
  - 検証用の関数Vがあり，(y,v)が正しく作成されたかどうか，すなわちV(y,v)=1かどうかを確認できる.
- SPK{(x,v):V(F(x),v)=1}(m)を署名とする.

関数F, S, V の設計が重要(難しい)

例(CS97)

- CRYPTO’97で発表されたグループ署名
  - 署名長が署名者集合Sのサイズに依存しない
  - ただし，署名長は短くはない.
- 数論問題
  - 離散対数問題と素因数分解問題に深く関連
    - double discrete logarithm of y to the bases g and a
    - e-th root of the discrete logarithm of y to the base g

  - RSA署名と離散対数問題に関する知識があれば理解できる. (本講義で取り扱う理由)

[CS97] Camenisch, Stadler, “Efficient Group Signature Schemes for Large Groups”, CRYPTO '97, LNCS 1294, pp. 410-424, 1997.

CS97で用いられた2つの数論問題

- double discrete logarithm of y to the bases g and a

$$\text{Given } P, n = pq, g, y, a \text{ where } n \mid P-1, |g| = n,$$

$$\text{find } \alpha \text{ s.t. } g^{(a^{\alpha})} \equiv y \pmod{P}$$

$$SPK\{(\alpha) : g^{(a^{\alpha})} \equiv y \pmod{P}\}(m)$$

これを利用

- e-th root of the discrete logarithm of y to the base g

$$\text{Given } P, n = pq, g, y, e \text{ where } n \mid P-1, |g| = n,$$

$$\text{find } \beta \text{ s.t. } g^{(\beta^{e})} \equiv y \pmod{P}$$

$$SPK\{(\beta) : g^{(\beta^{e})} \equiv y \pmod{P}\}(m)$$

これを利用

RSA署名(概要)

- 公開鍵(n,e)
  - 素数p,qの積n
  - gcd(e, ϕ(n))=1となるe
- 秘密鍵(d)
  - d = $e^{-1} \bmod \phi(n)$
- 署名 σ
  - メッセージMに対し，$\sigma = M^d \bmod n$
- 署名検証
  - メッセージと署名の対(M, σ)に対し，$M \stackrel{?}{\equiv} \sigma^e \bmod n$

管理者によるパラメタ生成

- 素数p,qの積nを計算(nはRSA署名で用いるn)
- n|P-1となる素数Pを計算
  - P=kn+1 (k=2,4,6,・・・)
- 位数がnとなるZ^*_Pの元gを計算

- グループ公開鍵
  - (n,P,g)
- 管理者の秘密鍵
  - p,q

各署名者の署名鍵の生成(登録処理)

- 署名者の秘密鍵x_i

- 署名者→管理者
  $$y_i = a^{x_i} \bmod n$$
  - 署名者はx_iの知識の証明を行う.
- 管理者→署名者
  $$v_i = (y_i + 1)^{1/e} \bmod n$$

署名者集合への登録処理

- 管理者以外は(nの素因数分解を知らないと), v_iを作成できそうにない.

グループ署名生成と検証

署名生成
$(\tilde{g},\tilde{z},\sigma_1,\sigma_2)$の生成

$$(\tilde{g},\tilde{z},\sigma_1,\sigma_2)where$$
$$\tilde{g}=g^r \bmod P(r\in_R Z_n^*),$$
$$\tilde{z}=\tilde{g}^y \bmod P$$
$$\sigma_1=SPK\{(\alpha):\tilde{z}\equiv\tilde{g}^{a^\alpha}(\bmod P)\}(m)$$
$$\sigma_2=SPK\{(\beta):\tilde{z}\tilde{g}\equiv\tilde{g}^{\beta^e}(\bmod P)\}(m)$$

署名検証
知識の署名
V_1,V_2の検証

σ_1,σ_2よりβ=$(a^\alpha+1)^{1/e} \bmod n$の知識の証明となる($\tilde{z}\tilde{g}=\tilde{g}^{a^\alpha+1}=\tilde{g}^{\beta^e}$だから).

署名者の開示

- y_iを所持している管理者は，全署名者のy_iに対して以下を計算

$$\tilde{z}=\tilde{g}^{y_i} \bmod P$$

- 計算で得た $\tilde{z}$ が署名$(\tilde{g},\tilde{z},\sigma_1,\sigma_2)$の第2成分と一致するかチェック

$$PK\{(\alpha): \tilde{z} = \tilde{g}^{a^{\alpha}}\}$$

P(証明者) V(検証者)

(秘密の情報x)

$a_i = \tilde{g}^{a^{r_i}} (i = 1, \cdots, l)$ $\xrightarrow{(a_1, \cdots, a_l)}$ $(b_1, \cdots, b_l): random$

$\xleftarrow{(b_1, \cdots, b_l)}$

$\begin{cases} c_i = r_i (b_i = 0) \\ c_i = r_i - x (b_i = 1) \end{cases}$ $\xrightarrow{(c_1, \cdots c_l)}$ $\begin{cases} a_i = \tilde{g}^{a^{c_i}} (b_i = 0) \\ a_i = \tilde{z}^{a^{c_i}} (b_i = 1) \end{cases}$

注1)$|r_i|>|x|$でなくてはならない.

注2)統計的ゼロ知識証明になる.

$$PK\{(\beta): \tilde{z}\tilde{g} = \tilde{g}^{\beta^e}\}$$

P(証明者) V(検証者)

(秘密の情報x)

$a_i = \tilde{g}^{r_i^e} (i = 1, \cdots, l)$ $\xrightarrow{(a_1, \cdots, a_l)}$ $(b_1, \cdots, b_l): random$

$\xleftarrow{(b_1, \cdots, b_l)}$

$\begin{cases} c_i = r_i (b_i = 0) \\ c_i = r_i / x \bmod n (b_i = 1) \end{cases}$ $\xrightarrow{(c_1, \cdots c_l)}$ $\begin{cases} a_i = \tilde{g}^{c_i^e} (b_i = 0) \\ a_i = (\tilde{z}\tilde{g})^{c_i^e} (b_i = 1) \end{cases}$

目標を達成できたか?(Informal)

- 署名 σ を署名者集合Sに含まれていない人が偽造できない.
  - 署名者集合Sへの登録処理を行わない限り, v_iを得ることはできない.
  - 署名にはv_iの知識の署名(σ_1, σ_2)が含まれている(つまり, v_iを得た人のみが署名生成可能).

$$v_i = (y_i + 1)^{1/e} \bmod n$$

- 署名 σ から署名者を特定できない.
  - v_iの知識の署名を作成しているのだが, (ゼロ知識証明なので)登録処理を行ったどのv_iを利用しているかという情報は漏れない.
- 特殊な管理者Ozzieのみは, 署名 σ から, 署名者が誰かを開示できる.
  - 署名者集合のサイズに比例するが, 開示は可能.
- 署名長, 公開鍵長は, 署名者集合のサイズに依存しない.

まとめ

- 匿名性を有する署名の構成例を2つ示した.
  - 1-out-of-n署名
    - 知識の署名の拡張(OR)により実現
    - 署名サイズが, 署名者集合Sのサイズに比例
  - グループ署名(CS97)
    - $\tilde{g}$ の指数部が(管理者からもらった)RSA署名
    - $\tilde{g}$ の指数部がRSA署名であることの知識の署名を構成
      - (管理者からもらった)RSA署名は, 検証者には見えない.
    - 署名サイズが, 署名者集合Sのサイズに依存しない