Anonymous and Transferable Electronic Ticketing Scheme

– Data Privacy Management, 8th International Workshop –

Arnau Vives-Guasch[1] M. Magdalena Payeras-Capellà[2]
Macià Mut-Puigserver[2] Jordi Castellà-Roca[1]
Josep-Lluís Ferrer-Gomila[2]

[1]Universitat Rovira i Virgili. Tarragona (Spain)

[2]Universitat de les Illes Balears. Mallorca (Spain)

Egham, UK. September 12-13, 2013.

Table of Contents

Introduction

- IT industry: smartphones revolution
  - Computation power
  - Storage capacity
  - Communication technologies (NFC, Wi-Fi, 4G, etc.)
  - Mobility+flexibility: payment and ticketing schemes
- Ticket: representation of the owner's rights to receive a determined service
  - At least, the same security requirements have to be fulfilled as in paper format
  - Requirements mainly depend on the service

Contribution

- E-ticketing system
- Group signatures
- Security requirements:
  - Anonymity (revocable)
  - Short-term linkability (adaptation from BBS scheme)
  - Transferability
- Easily deployable to real scenarios

Table of Contents

Security assumptions

Definition (The q-Strong Diffie-Hellman problem, SDH)

Given two cyclic groups G_1 and G_2 of prime order p, two randomly chosen generators $g_1 \in G_1$ and $g_2 \in G_2$ of their respective groups, with an isomorphism $\psi : G_2 \to G_1$ where $g_1 = \psi(g_2)$, the q-SDH problem is a hard computational problem where the $(q+2)$-tuple $(g_1, g_2, g_2^{\gamma}, g_2^{\gamma^2}, ..., g_2^{\gamma^q}) \in G_1 \times G_2^{q+1}$ is the input and the pair $(g_1^{\frac{1}{x+\gamma}}, x) \in G_1 \times \mathbb{Z}_p$ is the output, for some $x \in \mathbb{Z}_p^*$ such that $x + \gamma \neq 0$.

Definition (The Decision Linear Diffie-Hellman problem, DLIN)

Given a cyclic group G_1 of order p, and taking $u, v, h, u^a, v^b, h^c \in G_1$ as input, where $u, v, h \in G_1$ randomly chosen generators, and random $a, b, c \in \mathbb{Z}_p$, and output *yes* if $a + b = c$ and *no* otherwise.

Procedures

- BBS scheme:
  - $KeyGen_G$
  - $Sign_G$
  - $Verify_G$
  - $Open_G$
- ZKP of the BBS scheme:
  - $ZKP_G\,Commit$
  - $ZKP_G\,Response$
  - $ZKP_G\,Verify$
- Own adaptation for short-term linkability:
  - $SignLinkable_G$
  - $VerifyLinkable_G$

Procedures: $\mathbf{KeyGen_G(n)}$

Generate group of n users and their respective set of keys.

1. select $h \xleftarrow{R} G_1 \backslash \{1_{G_1}\}$
2. generate $gmsk = (\xi_1, \xi_2)$ where $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$
3. set $u, v \in G_1$ such that $u^{\xi_1} = v^{\xi_2} = h$
4. select $\gamma \xleftarrow{R} \mathbb{Z}_p^*$
5. set $w = g_2^{\gamma}$
6. generate $\forall \mathcal{U}_i$, $1 \leq i \leq n$, an SDH tuple (A_i, x_i) by:
   - select $x_i \xleftarrow{R} \mathbb{Z}_p^*$
   - set $A_i \leftarrow g_1^{1/(\gamma + x_i)}$
   - γ is the private master key of the group key issuer

Procedures: $\mathbf{Sign_G(gpk, gsk[i], M)}$ I

Given $gpk = (g_1, g_2, h, u, v, w)$, $gsk[i] = (A_i, x_i)$ and a message $M \in \{0,1\}^*$, output a signature of knowledge $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$.

1. select $\alpha, \beta \xleftarrow{R} \mathbb{Z}_p$
2. compute the linear encryption of A: $(T_1, T_2, T_3) \leftarrow (u^\alpha, v^\beta, Ah^{\alpha+\beta})$
3. compute $\delta_1 \leftarrow x\alpha$ and $\delta_2 \leftarrow x\beta$;
4. select $r_\alpha, r_\beta, r_x, r_{\delta_1}, r_{\delta_2} \xleftarrow{R} \mathbb{Z}_p$
5. compute:
   - $R_1 \leftarrow u^{r_\alpha}$
   - $R_2 \leftarrow v^{r_\beta}$
   - $R_3 \leftarrow e(T_3, g_2)^{r_x} \cdot e(h, w)^{-r_\alpha - r_\beta} \cdot e(h, g_2)^{-r_{\delta_1} - r_{\delta_2}}$
   - $R_4 \leftarrow T_1^{r_x} \cdot u^{-r_{\delta_1}}$
   - $R_5 \leftarrow T_2^{r_x} \cdot v^{-r_{\delta_2}}$
6. compute: $c \leftarrow H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5)$

Procedures: $\mathbf{Sign_G(gpk, gsk[i], M)}$ II

7. generate:
   - $s_\alpha \leftarrow r_\alpha + c\alpha$
   - $s_\beta \leftarrow r_\beta + c\beta$
   - $s_x \leftarrow r_x + cx$
   - $s_{\delta_1} \leftarrow r_{\delta_1} + c\delta_1$
   - $s_{\delta_2} \leftarrow r_{\delta_2} + c\delta_2$
8. output $\sigma \leftarrow (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$.

Procedures: $\mathbf{Verify_G(gpk, M, \sigma)}$

Given $gpk = (g_1, g_2, h, u, v, w)$, a message M and $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, verify that σ is a valid signature of the message

1. re-derive R_1, R_2, R_3, R_4, R_5:
   - $\tilde{R_1} \leftarrow u^{s_\alpha} / T_1^c$
   - $\tilde{R_2} \leftarrow v^{s_\beta} / T_2^c$
   - $\tilde{R_3} \leftarrow e(T_3, g_2)^{s_x} \cdot e(h, w)^{-s_\alpha - s_\beta} \cdot e(h, g_2)^{-s_{\delta_1} - s_{\delta_2}} \cdot (e(T_3, w)/e(g_1, g_2))^c$
   - $\tilde{R_4} \leftarrow T_1^{s_x} / u^{s_{\delta_1}}$
   - $\tilde{R_5} \leftarrow T_2^{s_x} / v^{s_{\delta_2}}$
2. verify $c \stackrel{?}{=} H(M, T_1, T_2, T_3, \tilde{R_1}, \tilde{R_2}, \tilde{R_3}, \tilde{R_4}, \tilde{R_5})$

Procedures: $\mathbf{Open_G(gpk, gmsk, M, \sigma)}$

- Trace a signature to a concrete signer inside the group
- $\mathcal{M}_\mathcal{G}$ holds *gmsk* master key and knows all (A_i, x_i) pairs
- Given $gpk = (g_1, g_2, h, u, v, w)$, $gmsk = (\xi_1, \xi_2)$, a message M and $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$:
  1. Recover user's identity: $A \leftarrow T_3/(T_1^{\xi_1} \cdot T_2^{\xi_2})$
  2. If elements $\{A_i\}$ of the $gsk[i]$ are given to $\mathcal{M}_\mathcal{G}$, look up the user index for A recovered from the signature

Procedures: $\mathbf{SignLinkable_G(gpk, gsk[i], M)}$

Given gpk, $gsk[i]$, a new message M', a previous signature σ, and the values α, β used for that signature, compute and output a signature σ'

- First use: standard $Sign_G(gpk, gsk[i], M)$. Obtains σ with (α, β)
- Further uses: $SignLinkable_G(gpk, gsk[i], M', \sigma, \alpha, \beta)$:
  1. use the same pair (α, β) producing the same linear encryption of A: $(T_1, T_2, T_3) = (u^{\alpha}, v^{\beta}, Ah^{\alpha+\beta})$
  2. given a message M', sign the message: $\sigma' \leftarrow (T_1, T_2, T_3, c', s'_{\alpha}, s'_{\beta}, s'_x, s'_{\delta_1}, s'_{\delta_2})$ where $c' \leftarrow H(M', T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5) \in \mathbb{Z}_p$

Procedures: $\mathbf{VerifyLinkable_G(\sigma, \sigma')}$

This algorithm takes two signatures σ and σ' as input and outputs *true* or *false* depending on whether the signatures have been produced by the same signer's pseudonym:

- $T_1 \stackrel{?}{=} T_1'$
- $T_2 \stackrel{?}{=} T_2'$
- $T_3 \stackrel{?}{=} T_3'$

Table of Contents

Requirements

- Authenticity
- Non-repudiation
- Integrity
- Revocable anonymity
- Short-term linkability
- Non-overspending
- Transferability

Participants

- User ($\mathcal{U}$)
- Issuer ($\mathcal{I}$)
- Service provider ($\mathcal{P}$)
- Group Manager ($\mathcal{M}_{\mathcal{G}}$)

Phases

- Ticket issue
- Ticket transfer
  - 1st time (from original)
  - Further times (from already transferred)
- Ticket verification
  - Standard (original)
  - Transferred
- Revocation of anonymity ($\mathcal{M}_{\mathcal{G}}$)

Phases: Ticket issue

User ($\mathcal{U}$)		**Issuer ($\mathcal{I}$)**
		$n_\alpha \xleftarrow{R} \mathbb{Z}_p$
	$\xleftarrow{n_\alpha}$	
selects Sv $\mathsf{V} = Sign_G(\mathsf{Sv}, n_\alpha, \mathsf{flag\_issue})$		
	$\xrightarrow{\mathsf{V}}$	
		$Verify_G(\mathsf{V})$ $\mathsf{T} = Sign_{\mathcal{I}}(\mathsf{Sn}, \mathsf{Sv}, \mathsf{Tc}, \mathsf{V}, ...)$
	$\xleftarrow{\mathsf{T}}$	
$Verify_{\mathcal{I}}(\mathsf{T})$		

Phases: Ticket transfer (1st time)

User sender ($\mathcal{U}_1$)		**User receiver ($\mathcal{U}_2$)**
$m_{\beta_0} = ZKP_G Commit(\mathsf{T})$		
	$\xrightarrow{m_{\beta_0}, \mathsf{T}}$	
		$Verify_{\mathcal{I}}(\mathsf{T})$ $Verify_G(\mathsf{T.V})$ $n_{\lambda_0} \xleftarrow{R} \mathbb{Z}_p$ $c_{\beta_0} = H(n_{\lambda_0}, k = 0, price, \text{flag\_transfer})$
	$\xleftarrow{c_{\beta_0}}$	
$s_{\beta_0} = ZKP_G Response(m_{\beta_0}, c_{\beta_0})$ $n_{\beta_0} \xleftarrow{R} \mathbb{Z}_p$		
	$\xrightarrow{s_{\beta_0}, n_{\beta_0}}$	
		$ZKP_G Verify(m_{\beta_0}, c_{\beta_0}, s_{\beta_0})$ $\mathsf{W}_0 = Sign_G(n_{\beta_0}, \mathsf{T}^*, \text{flag\_transfer});$
	$\xleftarrow{\mathsf{W}_0}$	
$Verify_G(\mathsf{W}_0)$ $\mathsf{X}_0 = SignLinkable_G(\mathsf{W}_0)$		
	$\xrightarrow{\mathsf{X}_0}$	
		$Verify_G(\mathsf{X}_0)$ $VerifyLinkable_G(\mathsf{T.V}, \mathsf{X}_0)$

Phases: Ticket transfer (further times)

User sender ($\mathcal{U}_1$)		**User receiver ($\mathcal{U}_2$)**
$m_{\beta_k} = ZKP_G\,Commit(\mathrm{X_{k-1}})$		
	$\xrightarrow{m_{\beta_k}, \mathrm{X_{k-1}}}$	
		$Verify_{\mathcal{I}}(\mathrm{T^*})$ $Verify_G(\mathrm{T.V})$ $VerifyLinkable_G(\mathrm{X_0, T.V})$ $\forall \mathbf{i} \in \mathbf{[0, k)} : \mathbf{\{}$ $Verify_G(\mathrm{X_i})$ $Verify_G(\mathrm{W_i})$ if $(\exists \mathrm{W_{i-1}})$ $VerifyLinkable_G(\mathrm{X_i, W_{i-1}})$ $\mathbf{\}}$
		$n_{\lambda_k} \xleftarrow{R} \mathbb{Z}_p$ $c_{\beta_k} = H(n_{\lambda_k}, k, price, \mathrm{flag\_transfer})$
	$\xleftarrow{c_{\beta_k}}$	
$n_{\beta_k} \xleftarrow{R} \mathbb{Z}_p$ $s_{\beta_k} = ZKP_G\,Response(m_{\beta_k}, c_{\beta_k})$		
	$\xrightarrow{s_{\beta_k}, n_{\beta_k}}$	
		$ZKP_G\,Verify(m_{\beta_k}, c_{\beta_k}, s_{\beta_k})$ $\mathrm{W_k} = Sign_G(n_{\beta_k}, \mathrm{X_{k-1}}, \mathrm{flag\_transfer})$
	$\xleftarrow{\mathrm{W_k}}$	
$Verify_G(\mathrm{W_k})$ $\mathrm{X_k} = SignLinkable_G(\mathrm{W_k, W_{k-1}})$		
	$\xrightarrow{\mathrm{X_k}}$	
		$Verify_G(\mathrm{X_k})$ $VerifyLinkable_G(\mathrm{X_k, W_{k-1}})$

Phases: Ticket verification (standard)

User ($\mathcal{U}$)	**Provider ($\mathcal{P}$)**
$\xrightarrow{\mathsf{T}^*}$	
	$Verify_{\mathcal{I}}(\mathsf{T}^*)$ $Verify_G(\mathsf{T.V})$ $n_\gamma \xleftarrow{R} \mathbb{Z}_p$
$\xleftarrow{n_\gamma}$	
$\mathsf{Y} = SignLinkable_G(n_\gamma, \mathsf{T.Sn}, \mathsf{flag\_spend\_standard})$	
$\xrightarrow{\mathsf{Y}}$	
	$Verify_G(\mathsf{Y})$ $VerifyLinkable_G(\mathsf{T.V}, \mathsf{Y})$

Phases: Ticket verification (transferred)

User ($\mathcal{U}$)		**Provider ($\mathcal{P}$)**
	$\xrightarrow{X_k}$	
		$Verify_{\mathcal{I}}(T^*)$ $Verify_G(T.V)$ $VerifyLinkable_G(X_0, T.V)$ $\forall \mathbf{i} \in \mathbf{[0, k]} : \{$ $Verify_G(X_i)$ $Verify_G(W_i)$ if $(\exists W_{i-1})$ $VerifyLinkable_G(X_i, W_{i-1})$ $\}$ $n_\gamma \xleftarrow{R} \mathbb{Z}_p$
	$\xleftarrow{n_\gamma}$	
$Y = SignLinkable_G(n_\gamma, T.Sn, \text{flag\_spend\_transferred})$		
	$\xrightarrow{Y}$	
		$Verify_G(Y)$ $VerifyLinkable_G(W_k, Y)$

Phases: Revocation of anonymity

In case of controversy (e.g. overspending), $\mathcal{M}_{\mathcal{G}}$ could revoke the anonymity of the misbehaver by calling $Open_G$ procedure

Table of Contents

Conclusions & future work

- Proposal for e-ticketing system
  - Revocable anonymity
  - Short-term linkability
  - Transferability
- Adaptation of group signature scheme for partial linkability
- Security analysis
- Future work:
  - Automated validation tools for the proposal
  - Performance analysis of current proposal in real scenario (mobile devices)
  - Atomic verification (chain of signatures)
  - Comparison of performance results

Anonymous and Transferable Electronic Ticketing Scheme

– Data Privacy Management, 8th International Workshop –

Arnau Vives-Guasch[1] M. Magdalena Payeras-Capellà[2]
Macià Mut-Puigserver[2] Jordi Castellà-Roca[1]
Josep-Lluís Ferrer-Gomila[2]

[1]Universitat Rovira i Virgili. Tarragona (Spain)

[2]Universitat de les Illes Balears. Mallorca (Spain)

Egham, UK. September 12-13, 2013.