Fair Traceable Multi-Group Signatures

Vicente Benjumea[1], Seung Geol Choi[2], Javier Lopez[1] and Moti Yung[3]

[1] Computer Science Dpt. University of Malaga. Spain

[2] Computer Science Dpt. Columbia University. USA

[3] Google Inc. & Computer Science Dpt. Columbia University. USA

Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

Group Signatures and Alike (I)

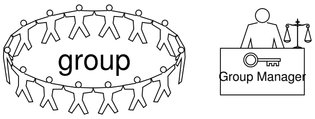
Group Signatures [CvH91, ACJT00]

- Crypto primitive supporting anonymity in different scenarios

Group Signatures and Alike (I)

Group Signatures [CvH91, ACJT00]

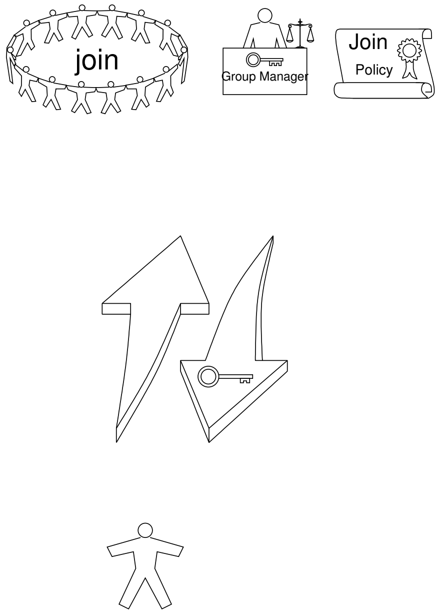
- Crypto primitive supporting anonymity in different scenarios
- GroupSetup: creation of a group



Group Signatures and Alike (I)

Group Signatures [CvH91, ACJT00]

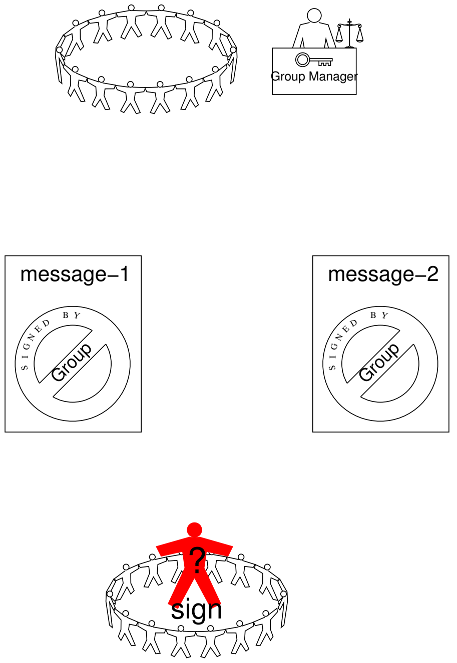
- Crypto primitive supporting anonymity in different scenarios
- GroupSetup: creation of a group
- Join: join to group



Group Signatures and Alike (I)

Group Signatures [CvH91, ACJT00]

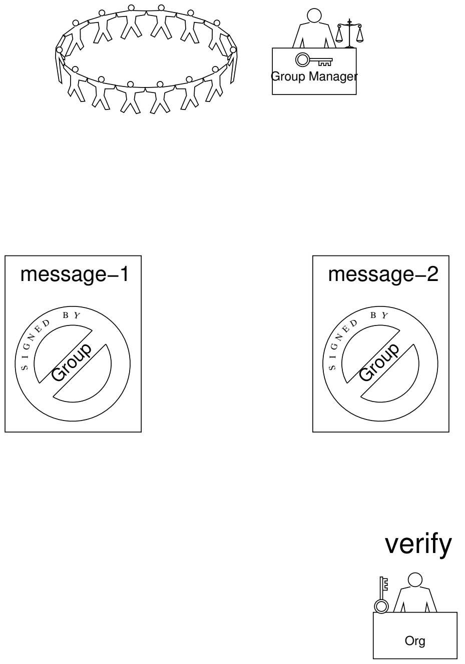
- Crypto primitive supporting anonymity in different scenarios
- GroupSetup: creation of a group
- Join: join to group
- Sign: issue a group sign. (anon&unlink)



Group Signatures and Alike (I)

Group Signatures [CvH91, ACJT00]

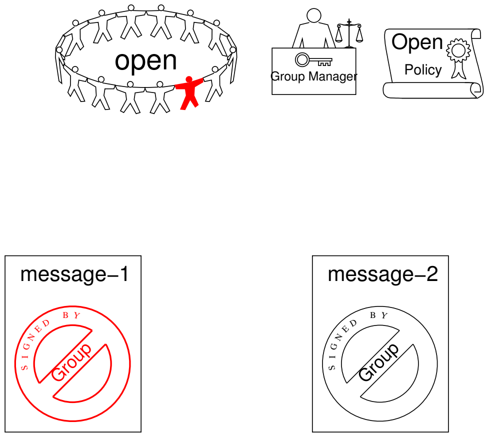
- Crypto primitive supporting anonymity in different scenarios
- GroupSetup: creation of a group
- Join: join to group
- Sign: issue a group sign. (anon&unlink)
- Verify: verify a group sign. (anon&unlink)



Group Signatures and Alike (I)

Group Signatures [CvH91, ACJT00]

- Crypto primitive supporting anonymity in different scenarios
- GroupSetup: creation of a group
- Join: join to group
- Sign: issue a group sign. (anon&unlink)
- Verify: verify a group sign. (anon&unlink)
- Open: identify the issuing member



Group Signatures and Alike (II)

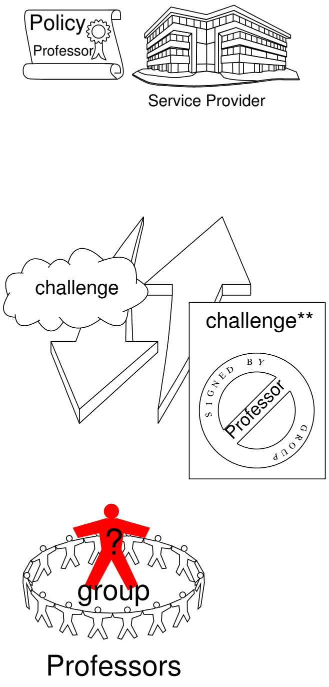
Group Signatures
(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity

Group Signatures and Alike (II)

Group Signatures
(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group

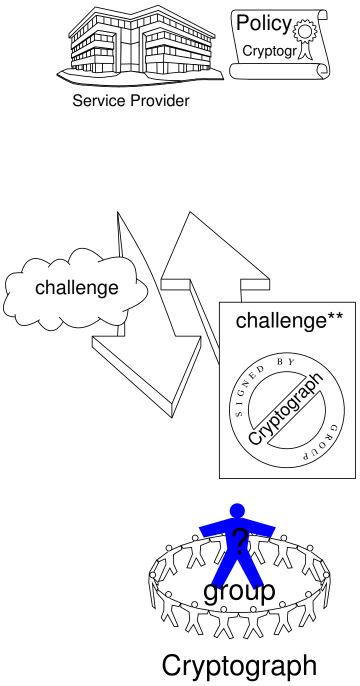


Group Signatures and Alike (II)

Group Signatures

(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group
- Anonymous auth within crypto group

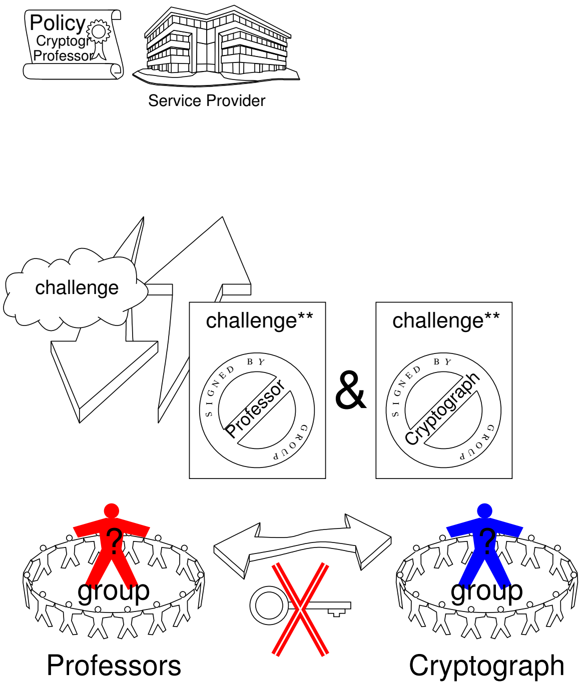


Group Signatures and Alike (II)

Group Signatures

(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group
- Anonymous auth within crypto group
- Simultaneous auth within both groups What guarantees the SP that both auths belong to the same anonymous user?

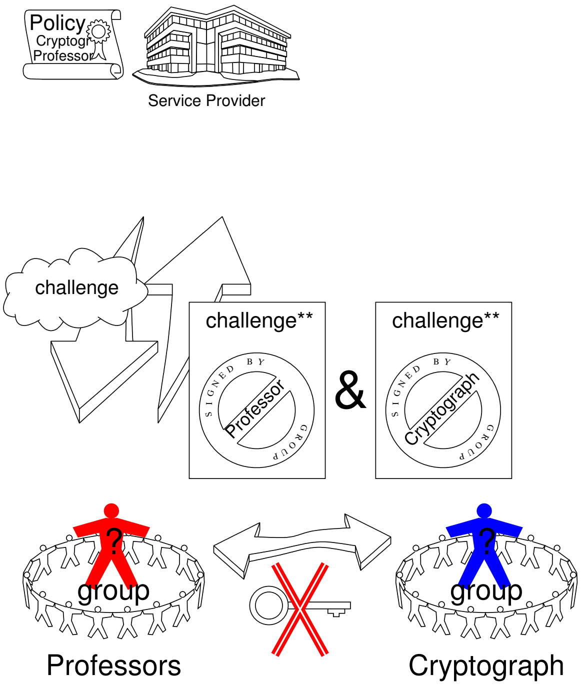


Group Signatures and Alike (II)

Group Signatures

(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group
- Anonymous auth within crypto group
- Simultaneous auth within both groups What guarantees the SP that both auths belong to the same anonymous user?
- Multi-group signatures [AT99] guarantee that two group signatures have been issued by the same anonymous user

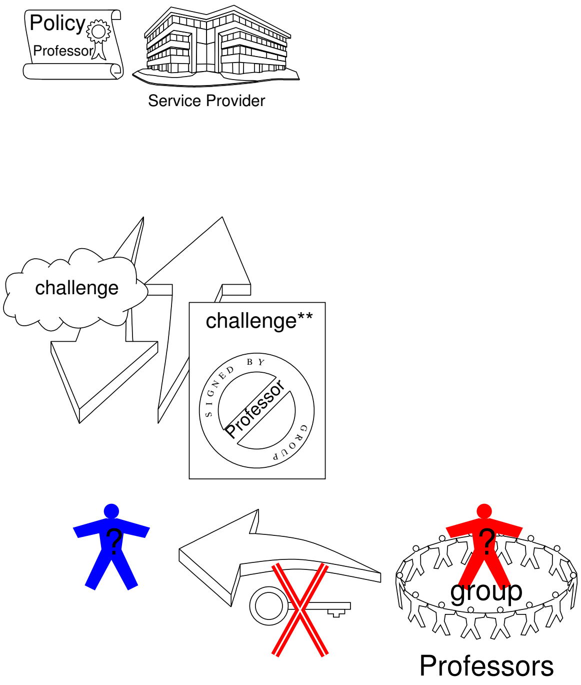


Group Signatures and Alike (II)

Group Signatures

(Authentication & Authorization)

- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group
- Anonymous auth within crypto group
- Simultaneous auth within both groups What guarantees the SP that both auths belong to the same anonymous user?
- Multi-group signatures [AT99] guarantee that two group signatures have been issued by the same anonymous user
- Additionally, users may decide to share **some** of the private keys

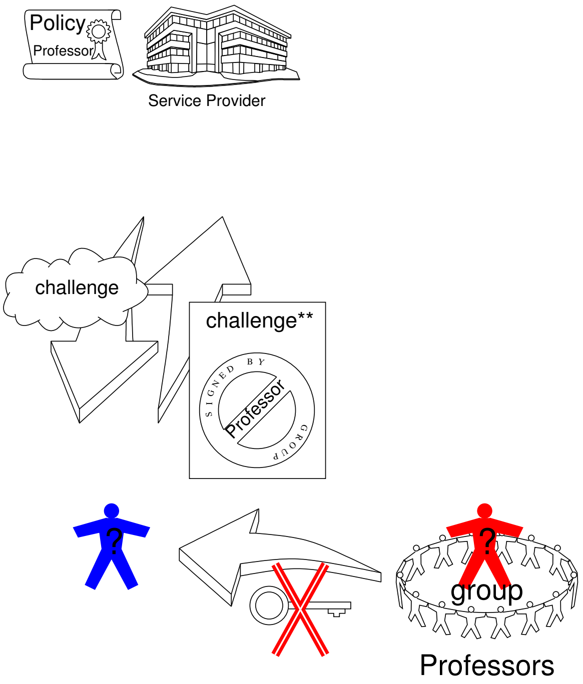


Group Signatures and Alike (II)

Group Signatures

(Authentication & Authorization)

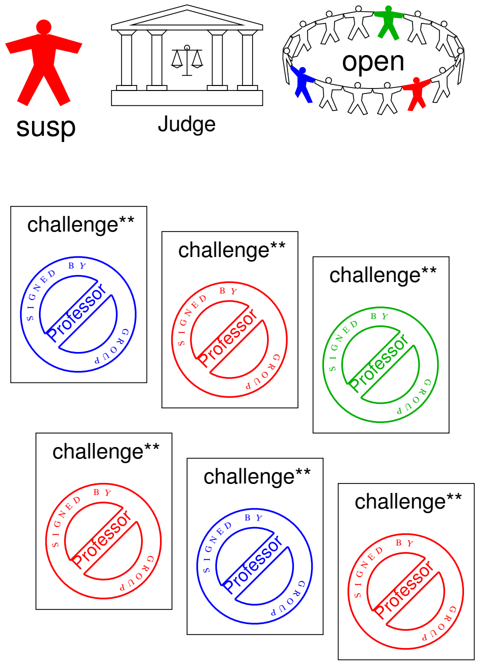
- In authentication & authorization scenarios, group signatures provide a suitable support for anonymity
- Anonymous auth within professors group
- Anonymous auth within crypto group
- Simultaneous auth within both groups
  What guarantees the SP that both auths belong to the same anonymous user?
- Multi-group signatures [AT99] guarantee that two group signatures have been issued by the same anonymous user
- Additionally, users may decide to share **some** of the private keys
- Embedding some valuable information into private keys may deter this sharing [DLN96, LRSW99]



Group Signatures and Alike (III)

Group Signatures

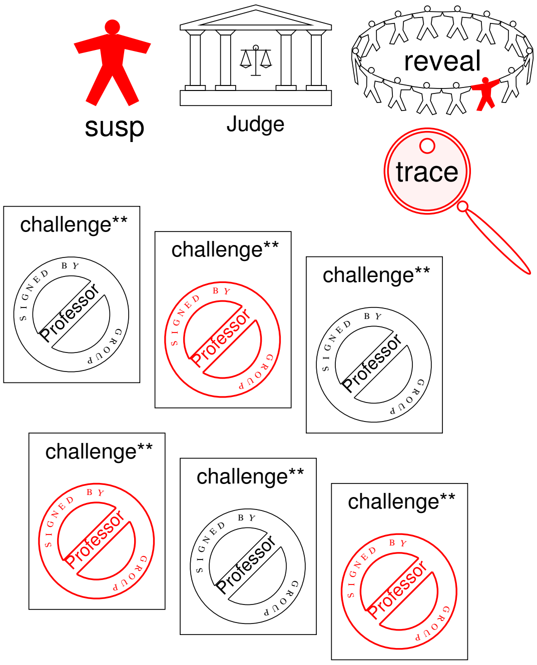
- When a user is under suspicion, the group manager can open the group signatures to see which ones were issued by that user
- However this approach violates other members' privacy



Group Signatures and Alike (III)

Group Signatures

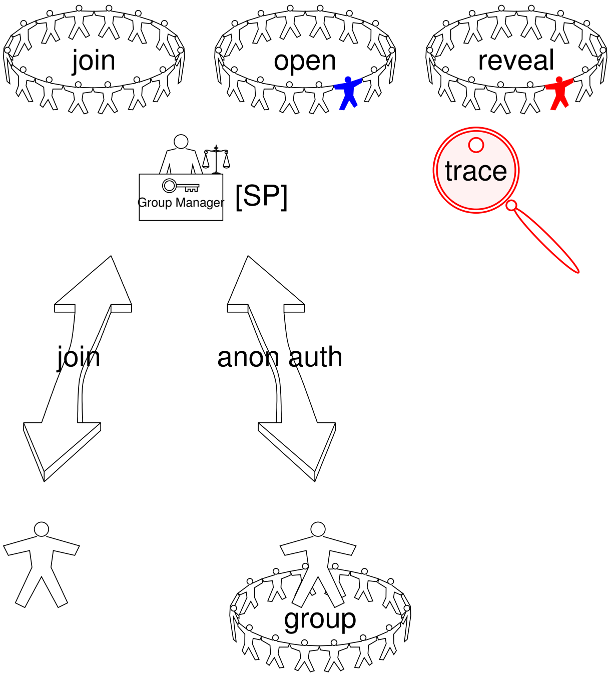
- When a user is under suspicion, the group manager can open the group signatures to see which ones were issued by that user
- However this approach violates other members' privacy
- Traceable signatures [KTY04] incorporate a tracing facility to identify the signatures issued by a given member, but respecting other members' privacy
- Additionally, a member is also able to claim authorship for a given signature


susp
Judge
reveal
trace
challenge**
SIGNED BY Professor GROUP
challenge**
SIGNED BY Professor GROUP
challenge**
SIGNED BY Professor GROUP
challenge**
SIGNED BY Professor GROUP
challenge**
SIGNED BY Professor GROUP
challenge**
SIGNED BY Professor GROUP

Group Signatures and Alike (IV)

Group Signatures

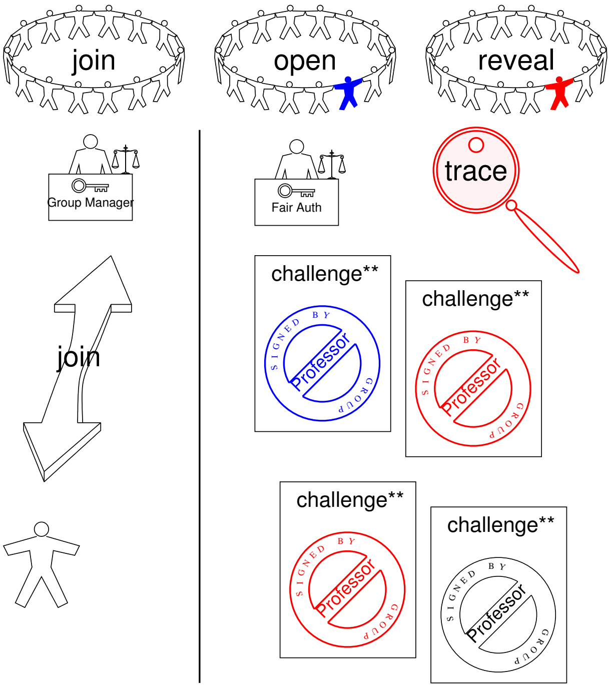
- The group manager is able to open a signature and identify the member that issued it
- Additionally, in traceable signatures, the group manager is able to trace the signatures issued by a given member
- What happens if the SP that provides a service is the GM itself ?
- What happens if the GM is a party in interest ? (it is not trusted with respect to users privacy)



Group Signatures and Alike (IV)

Group Signatures

- The group manager is able to open a signature and identify the member that issued it
- Additionally, in traceable signatures, the group manager is able to trace the signatures issued by a given member
- What happens if the SP that provides a service is the GM itself ?
- What happens if the GM is a party in interest ? (it is not trusted with respect to users privacy)
- The original roles of the group manager should be divided (Join vs. Open/Reveal/Trace) [KY04]



Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

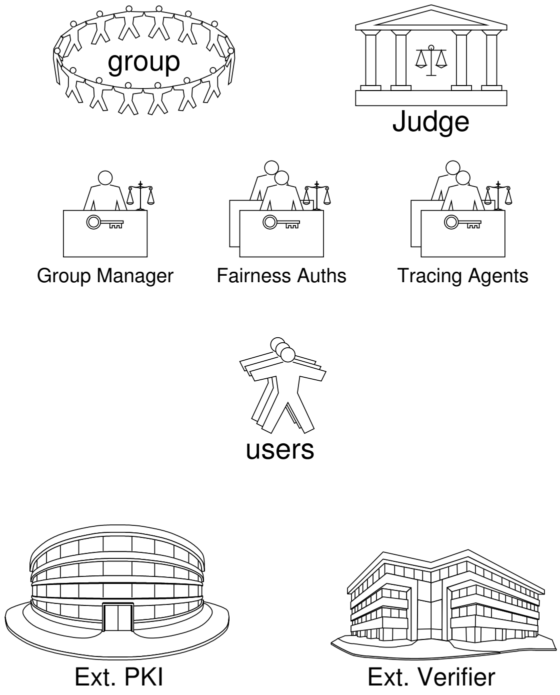
Our Main Goal

- Define an anonymous signature scheme concerned with previous scenarios
  - **Anonymous & unlinkable** signatures in the same way as Group and Traceable signatures
  - **Multi-group** features provide the guarantee that several signatures have been issued by the same anonymous user
  - Includes a mechanism to **dissuade** the group members from sharing the private keys.
  - **Splits** the original duties of the group manager
    - Group manager: **joins** new members to the group
    - Fairness authorities: **open** signatures and **reveal** tracing trapdoors.

Fair Traceable Multi-Group Signatures (FTMGS) (pronounced FaT-MuGS) (I)

Participating entities

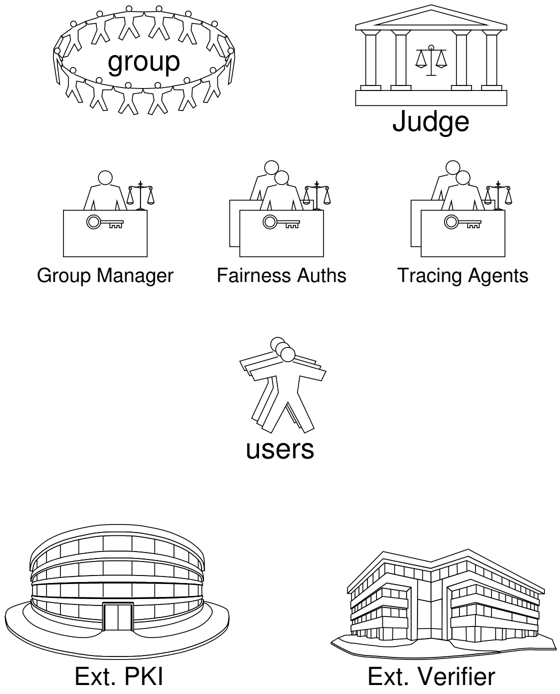
- Group manager (GM)
- Multiple fairness authorities (FA)
- Multiple tracing agents (TA)
- Judge (J)
- Multiple users (U)
- External verifiers (V)
- External PKI



Fair Traceable Multi-Group Signatures (FTMGS) (II)

Operations

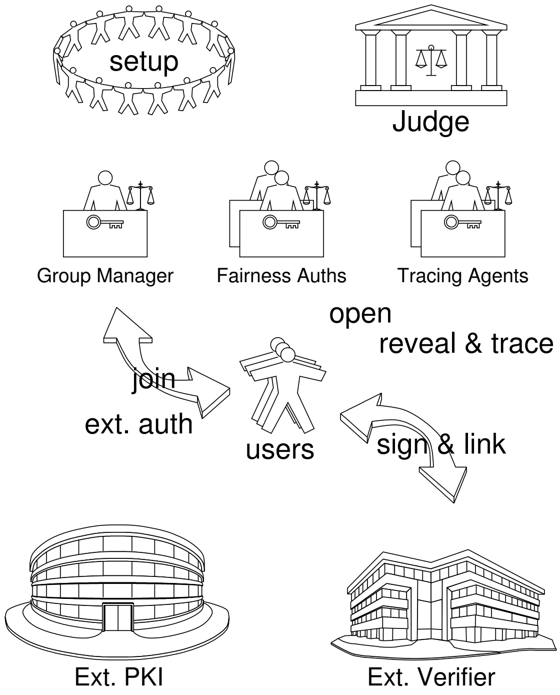
- Group setup
- JoinOnAuth
- Sign / Verify
- Open
- Reveal
- Trace
- Claim / Verify
- ClaimLink / Verify



Fair Traceable Multi-Group Signatures (FTMGS) (III)

General Scenario

- The GM **creates** the group with the collaboration of the FAs
- The user **joins** the group (external authorization)
- For a given transaction, the user **issues signatures** and **link** them (the membership proof is fair)
- Under critical circunstances, the judge, GM and FAs collaborate to: (breaking anonymity is also fair)
  - **Open** a signature
  - **Reveal** a tracing trapdoor that TAs use to **trace** member's signatures
- In some cases, a member can **claim** authorship for a given signature



Fair Traceable Multi-Group Signatures (FTMGS) (IV)

- When the user joins the group, she has been previously (and externally) authorized to do so
- The user is forced to embed her **master key** into her membership private keys.
  - This **master key** is the private key corresponding to her public key (PKI)
  - **Dissuades** users from sharing their membership private keys
  - Signatures can be **linked** by proving that they have been issued by membership private keys into which the same master key is embedded
  - Makes possible that a user can **link** inter-group signatures
  - Different users have different master keys
  - Signatures from different users **can not be linked**
  - Integrates **non-repudiation** into the scheme
  - It allows both, **identified** as well as **anonymous** join
- Linking signatures is under the user's control

Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

Fair Traceable Multi-Group Signatures: Construction of the Scheme

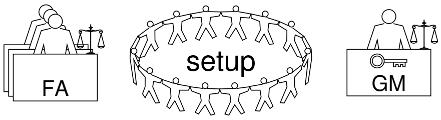
- System Parameters

Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
  - The security parameter ν
  - $\epsilon \in \mathbb{R}$ such that $\epsilon > 1$
  - $k \in \mathbb{N}$
  - Three spheres $\mathsf{\Lambda}$, M, $\mathsf{\Gamma}$,
  - Three inner spheres $\mathsf{\Lambda}_{\epsilon}^{k}$, $\mathsf{M}_{\epsilon}^{k}$, $\mathsf{\Gamma}_{\epsilon}^{k}$
- Signatures of Knowledge
  - Fiat-Shamir transformation [FS86] of interactive proof of knowledge into non-interactive in the random oracle model
  - Notation: $\mathsf{SK}\{(a,b) : y = g^a \ ; \ z = h^a f^b\}(m)$

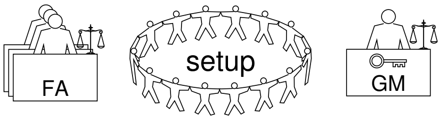
Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup



FAs: generate public RSA modulus $\widehat{n}$ with unknown fact. DKGP [FS01]

FA$_0$: selects $\widehat{g}' \in_R \mathbb{Z}_{\widehat{n}^2}$ and sets $\widehat{g} = \widehat{g}'^{2\widehat{n}}$

FA$_j$: selects a random prime $\widehat{o}_j \in_R \mathbb{Z}_{\widehat{n}^2/4}$, and computes $\widehat{y}_j = \widehat{g}^{\widehat{o}_j}$

GM: selects $n = pq$, $a_0, a, b, g \in_R QR(n)$, s.t. $p = 2p' + 1$, $q = 2q' + 1$ primes

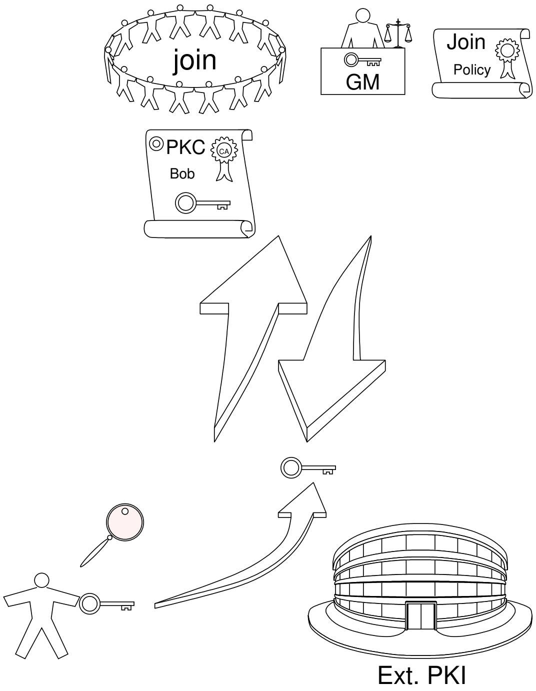
FA$_j$: selects $h_j \in_R QR(n)$, a random prime $o_j \in_R \mathbb{Z}_{\nu/2}$, and computes $y_j = g^{o_j}$

GM: computes $h = \prod_{j=1}^{\zeta} h_j$, $y = \prod_{j=1}^{\zeta} y_j$, $\widehat{y} = \prod_{j=1}^{\zeta} \widehat{y}_j$

GPK: $\langle\ n, a_0, a, b, g, h, y, \widehat{n}, \widehat{g}, \widehat{y}\ \rangle$

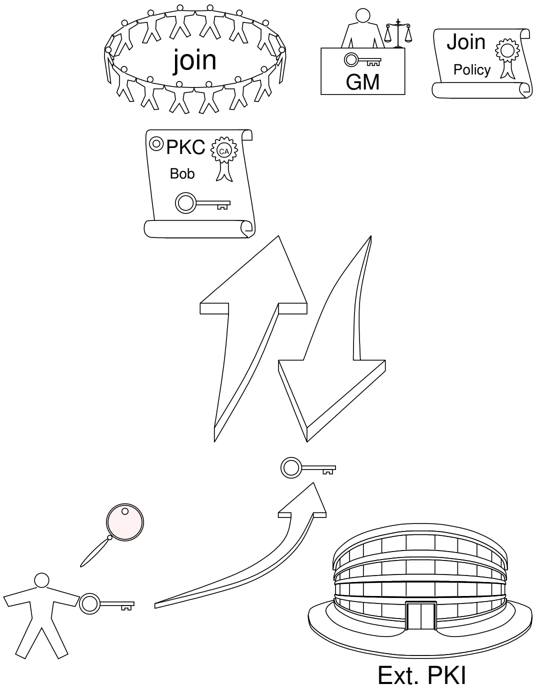
Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth



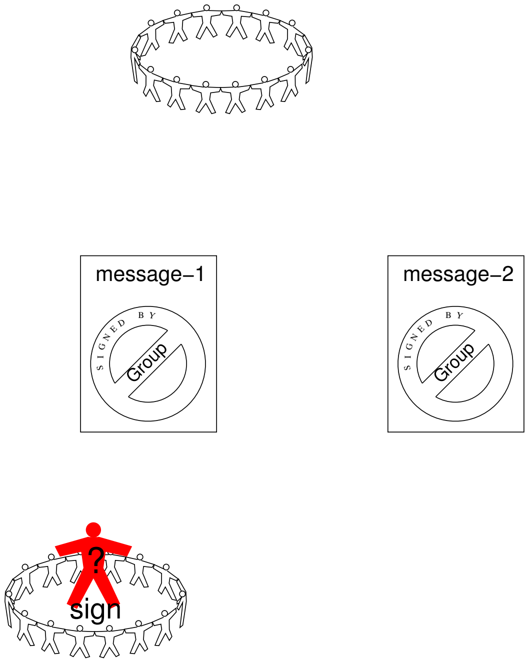
Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
  - Inputs: GPK, β, γ
    U: umk$_u$ = dlog$_\beta(\gamma)$ GM: p, q
  - [x'_i = umk$_u$] U→GM [$C_i = b^{x'_i}$]
  - [$x_i \in_R \Lambda^k_\epsilon$] U↔GM [$X_i = a^{x_i}$] [KTY04]
  - U→GM [$E_i = \langle U_i = \widehat{g}^{\widehat{r}}, V_i = \widehat{y}^{\widehat{r}} \widehat{h}^{x_i} \rangle$]
  - U→GM [SK$\{(x', r, x) : C_i = b^{x'}; \gamma = \beta^{x'}$
    $X_i = a^x; U_i = \widehat{g}^r; V_i = \widehat{y}^r \widehat{h}^x\}(\cdot)$]
  - [$e_i, A_i = (C_i X_i a_0)^{e_i^{-1}}$] U←GM [$e_i \in_R \Gamma^k_\epsilon$]
  - Outputs:
    U: $\langle A_i, e_i, x_i, x'_i \rangle$
    GM: $\langle A_i, e_i, C_i, X_i, U_i, V_i, \gamma, \beta,$ SK $\rangle$



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign

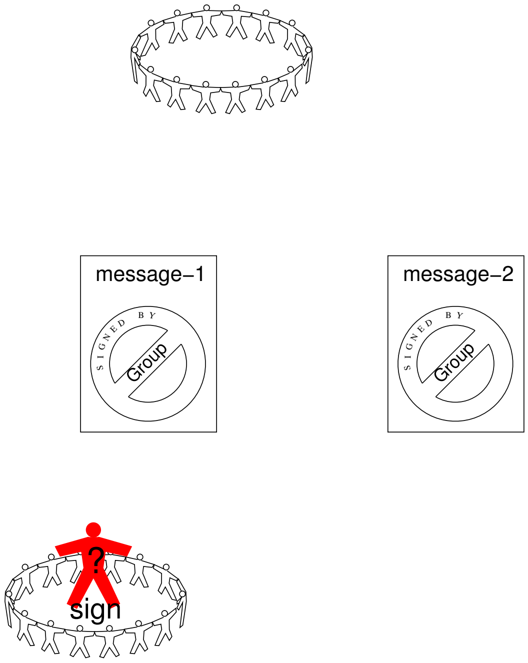


Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign

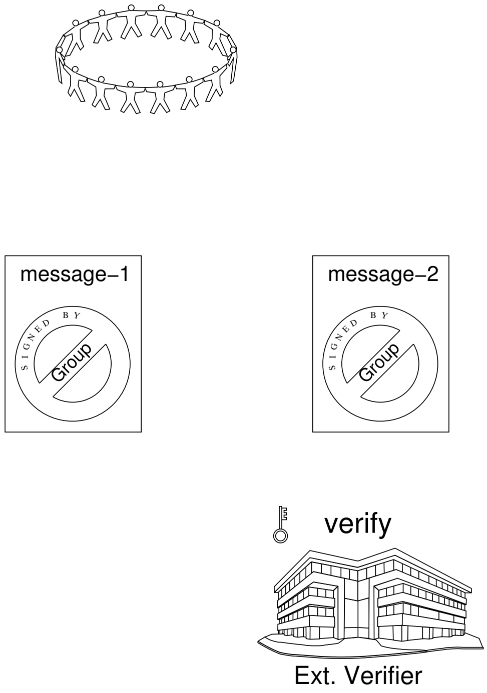
U: $T_1 = A_i y^r,\ T_2 = g^r,\ T_3 = g^{e_i} h^r,$
$T_4 = g^{x_i k},\ T_5 = g^k,$
$T_6 = g^{x_i' k'},\ T_7 = g^{k'}$

U: SK$\{(x, x', e, r, h'):$
$T_2 = g^r;\ T_3 = g^e h^r;$
$T_2^e = g^{h'};\ T_5^x = T_4;$
$T_7^{x'} = T_6;\ a_0 a^x b^{x'} y^{h'} = T_1^e\}(msg)$



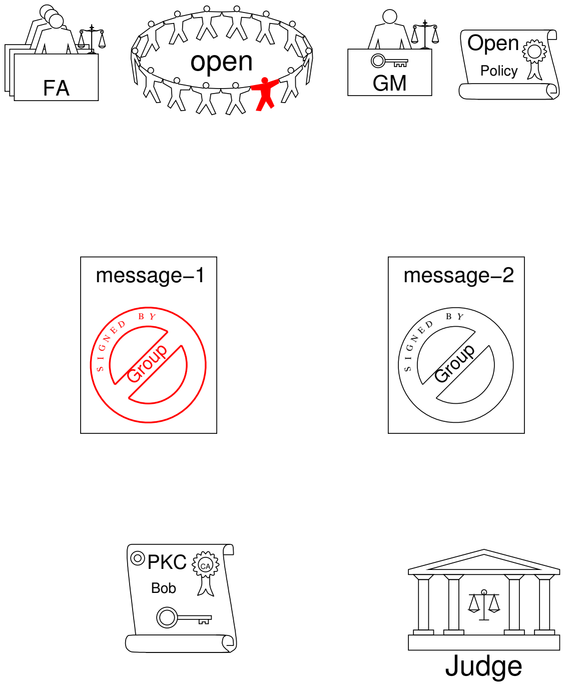
Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify (verifies signature of knowledge)



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature



Fair Traceable Multi-Group Signatures: Construction of the Scheme

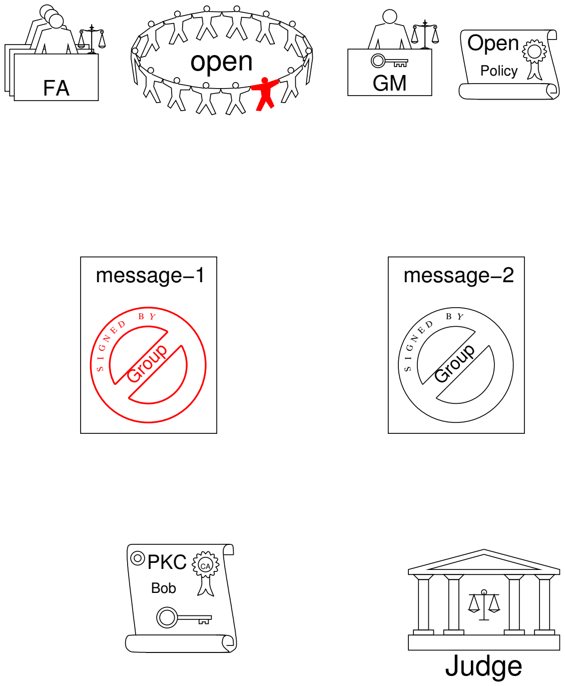
- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature

σ contains: $T_1 = A_i y^r, \; T_2 = g^r$

FA$_j$: computes $\widehat{\omega}_{j\sigma} = T_2^{o_j}$,
SK$\{(o) : y_j = g^o \; ; \; \widehat{\omega}_{j\sigma} = T_2^o\}(\sigma)$

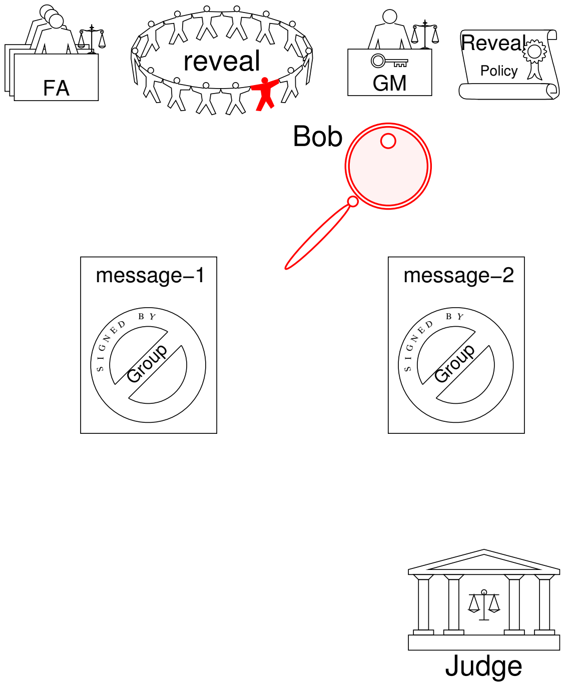
J: computes $\omega_\sigma = T_1 / (\prod_{j=1}^{\zeta} \widehat{\omega}_{j\sigma})$

GM: compares ω_σ with A_i in DB



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor



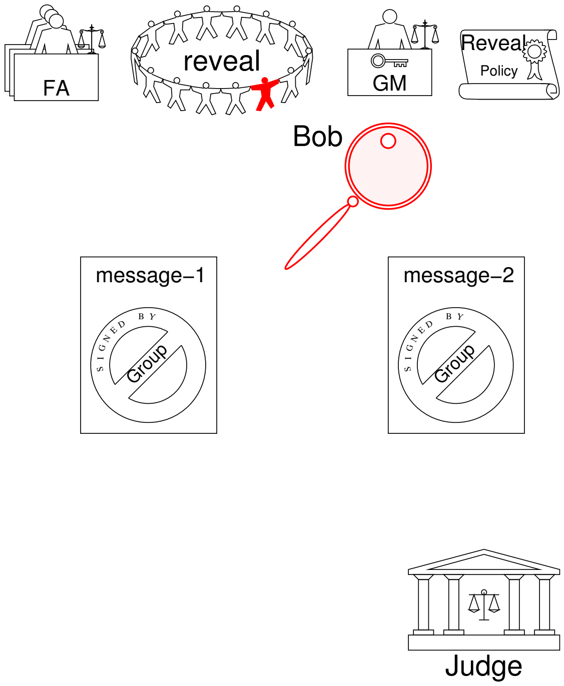
Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor

GM: knows (join) $U_i = \widehat{g}^{\widehat{r}}, V_i = \widehat{y}^{\widehat{r}}\widehat{h}^{x_i}$

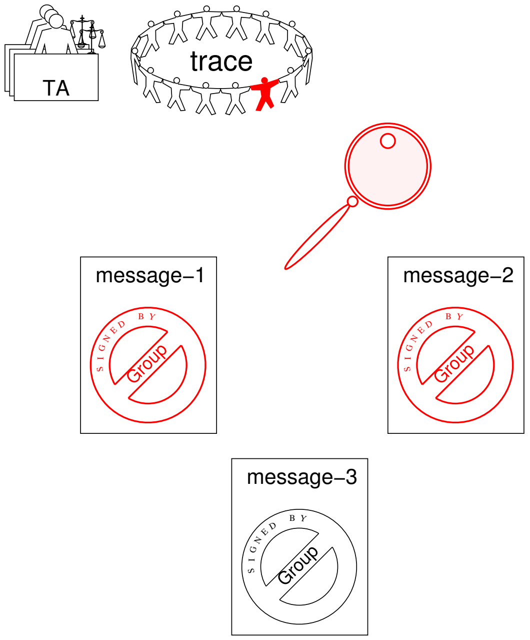
FA$_j$: computes $\widehat{\tau}_{ji} = U_i^{\widehat{o}_j}$
SK$\{(o) : \widehat{y}_j = \widehat{g}^o \ ; \ \widehat{\tau}_{ji} = U_i^o\}$(jlog$_i$)

J: computes $t = 2^{-1}$, and
$\widehat{x}_i = (V_i/(\prod_{j=1}^{\zeta} \widehat{\tau}_{ji}))^{2t}$, $\tau_i = (\widehat{x}_i - 1)/\widehat{n}$



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures

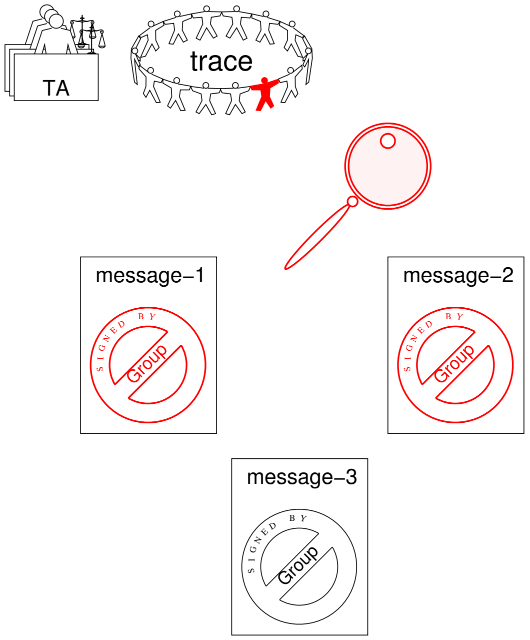


Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures

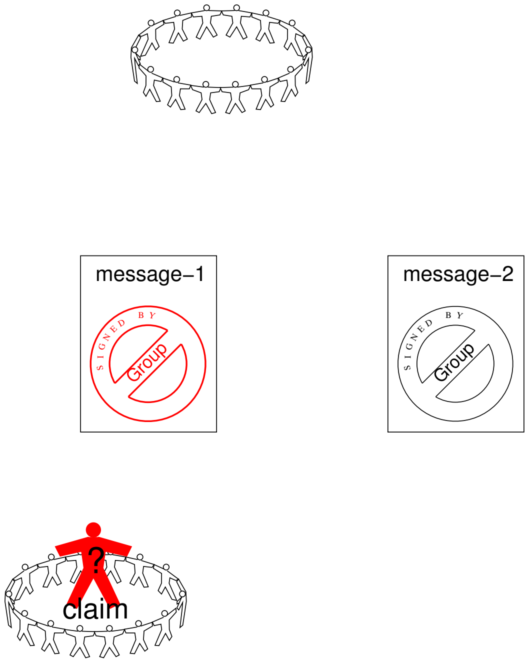
σ contains: $T_4 = g^{x_i k}$, $T_5 = g^k$

TA$_j$: checks if $T_4 = T_5^{\tau_i}$



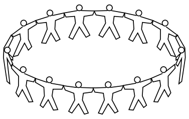
Fair Traceable Multi-Group Signatures: Construction of the Scheme

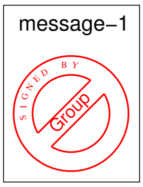
- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship

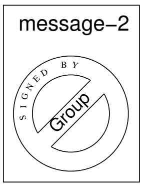


Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship





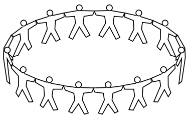


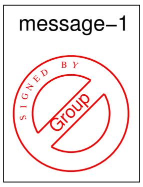
σ contains: $T_6 = g^{x_i' k'}$, $T_7 = g^{k'}$

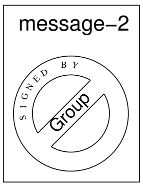
U: computes SK$\{(x') : T_6 = T_7^{x'}\}(\sigma, \gamma)$

Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship
- VerifyClaim (verifies sign. of knowledge)

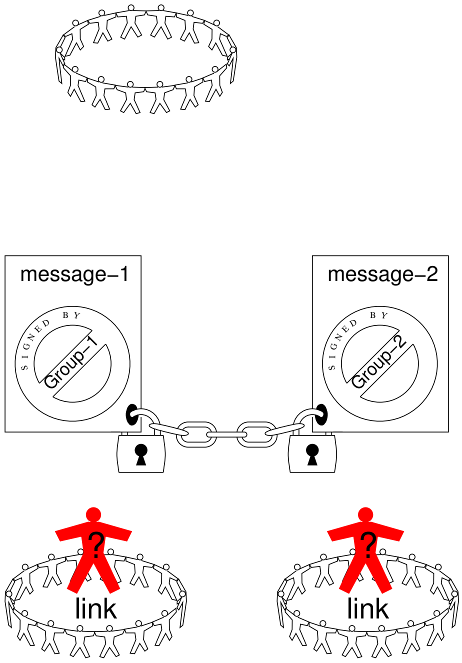






Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship
- VerifyClaim
- ClaimLink

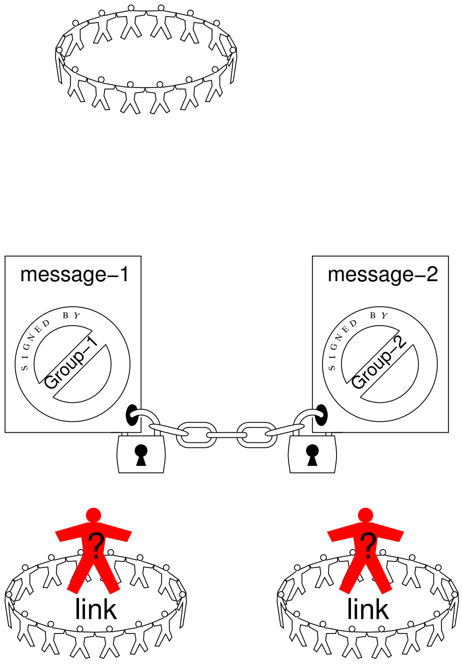


Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship
- VerifyClaim
- ClaimLink

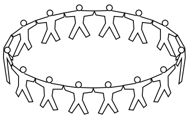
σ contains: $T_6 = g^{x'_i k'}$, $T_7 = g^{k'}$

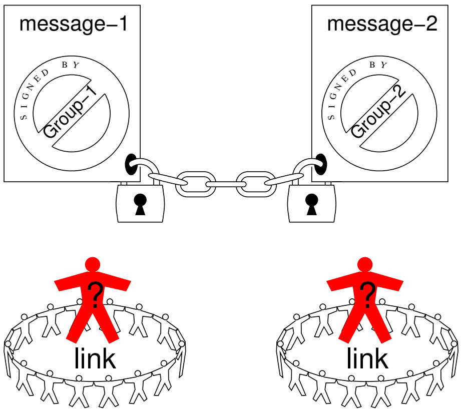
U: computes SK$\{(x') : T_{6\sigma_1} = T_{7\sigma_1}^{x'}$;
$T_{6\sigma_2} = T_{7\sigma_2}^{x'}\}(\sigma_1, \sigma_2, \gamma)$



Fair Traceable Multi-Group Signatures: Construction of the Scheme

- System Parameters
- Group-Setup
- JoinOnAuth
- Sign
- Verify
- Open a signature
- Reveal a tracing trapdoor
- Trace signatures
- Claim authorship
- VerifyClaim
- ClaimLink
- VerifyLink (verifies sign. of knowledge)





Fair Traceable Multi-Group Signatures: JoinOnAuth Scenario

- Join scenario. If the user has been autorized to join the group:
  - Either was identified, then the user's public key (DSA) is used for $\langle\ \beta, \gamma\ \rangle$ such that her private key is the user's master key [$\mathsf{umk}_u = \mathsf{dlog}_\beta(\gamma)$].
  - or was anonymously authenticated, for which issued a FTMGS, then the pair $\langle\ T_6, T_7\ \rangle$ from the signature is used for $\langle\ \beta, \gamma\ \rangle$ such that the user's master key remains constant [$\mathsf{umk}_u = \mathsf{dlog}_{T_6}(T_7)$].
- Note that non-repudiation also holds even in multiple-chained anonymous joins

Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

Security

Misidentification attack: the adversary tries to produce a signature that does not open or trace to any of the adversarially controlled users

Framing attack: the adversary tries to generate a signature, claim or link-claim that traces to a honest user

Anonymity attack: the adversary tries to break the anonymity of signatures

Link-forgery attack: the adversary tries to forge a false link

Join-anonymity attack: the adversary tries to track a member's joining situation

Security (in the random oracle model)

Misidentification attack: Strong-RSA [BP97]

Framing attack: Discrete-Logarithm & Decision Composite Residuosity [P99]

Anonymity attack: Decisional Diffie-Hellman [KTY04] & Decision Composite Residuosity [P99]

Join-anonymity attack: Cross Group DDH [JJN02]

Link-forgery attack: Strong-RSA [BP97]

Security Model and Proofs: are detailed in a full paper in eprint archive

`http://eprint.iacr.org/2008/047`

Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

Performance Analysis

	ACJT00	CL01	FTMGS
Member-Size (bytes)	1280	608	1488
Sign-Size (bytes)	656	1728	1312
Sign-Exp	12	28	21
Vrfy-Exp	11	30	21

Summary of Features

	ACJT00	CL01	FTMGS
Anonymous	+	+	+
Unlinkable	+	$-^{(\star)}$	+
Reversible	+	+	+
Traceable	−	−	+
Revocable	−	$-^{(\ddagger)}$	+
MultiGroup	−	$+^{(\star)}$	+
DeterSharing	−	+	+
Fairness	−	+	+
Non-Repudiation	$+^{(\dagger)}$	+	+

Agenda

1. Group Signatures and Alike
2. Fair Traceable Multi-Group Signatures (FTMGS)
3. Construction of the Scheme
4. Security
5. Performance Analysis
6. Conclusions

Conclusions

- We have presented Fair Traceable Multi-Group Signatures (FTMGS)
- It combines features from group / traceable signatures and multi-group signatures
- It also incorporates a mechanism to dissuade users from sharing their private keys
- Introduces a threshold scheme to guarantee fairness in opening and tracing signatures.
- The scheme is quite suitable to support anonymity in real world scenarios
- The new signature scheme can also be incorporated into a standard framework (X.509, SPKI) to support anonymous authentication/authorization [BCLY07]

Thank you for your attention

QUESTIONS ?