Traceable Group Encryption

Benoît Libert[1] **Moti Yung**[2] **Marc Joye**[1] **Thomas Peters**[3]

[1]Technicolor (France)
[2]Google Inc. and Columbia University (USA)
[3] UCL Crypto Group (Belgium)

March 28, 2014
Buenos Aires

Outline

Group Encryption

Kiayias-Tsiounis-Yung (Asiacrypt'07): encryption analogue of group signatures.

- Involves a group manager (GM) and an opening authority (OA).
- Sender CCA2-encrypts a message for a (certified) group member who remains anonymous in the CCA2-sense ...
- ...and generates a proof that
  - the ciphertext is valid and intended for some certified group member
  - the OA will be able to identify the receiver
  - the plaintext is a witness satisfying some relation

Group Encryption

- Applications:
  - Sender can encrypt emails to anonymous organization members while appending proofs that the content is not a spam/malware
  - Verifiable encryption of messages/keys to anonymous TTP

    ex.: International escrow system where users may prefer hiding their preferred TTP
  - Oblivious retriever storage: server temporarily stores encrypted data for anonymous retrievers

    ex.: Asynchronous transfers of encrypted credentials / datasets via the cloud
  - Group signatures with *ad-hoc* opening, hierarchical group signatures

Group Encryption

- Related work:
  - Kiayias-Tsiounis-Yung (Asiacrypt'07):
    - Modular design from key-private public key encryption, digital signatures, extractable commitments and ZK proofs
    - Efficient construction from Paillier;
      Proofs require either interaction or the ROM
  - Qin *et al.* (Inscrypt'08): related primitive with better efficiency in the ROM under interactive assumptions
  - Cathalo-Libert-Yung (Asiacrypt'09): construction with non-interactive proofs in the standard model
  - Izabachène-Pointcheval-Vergnaud (Latincrypt'10): individual users' traceability; removal of subliminal channels
  - El Aimani-Joye (ACNS'13): optimized constructions with interactive or non-interactive proofs

Group Encryption

- Almost all previous constructions require to open all ciphertexts to find those encrypted for a *specific group member*
  - Damaging to the privacy of well-behaved users
  - Tracing is an inherently sequential operation
- Exception: Izabachène-Pointcheval-Vergnaud (Latincrypt'10) gives individual traceability, but without explicit opening and only with IND-CPA security

  $\Rightarrow$ Explicitly "opening" one ciphertext in a population of n users requires $O(n)$ operations
- Need for a mechanism, akin to traceable signatures (Kiayias-Tsiounis-Yung, Eurocrypt'04), allowing to individually trace users
- **This paper**: primitive named Traceable Group Encryption, encryption analogue of traceable signatures

Traceable Group Encryption

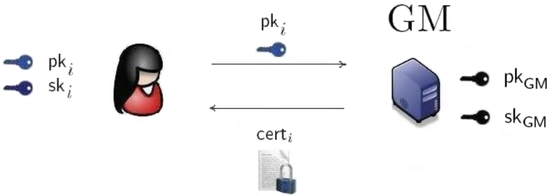
Properties:

- Encryption analogue of traceable signatures (Kiayias-Tsiounis-Yung, Eurocrypt'04)
- Opening authority can release a user-specific trapdoor allowing to trace all ciphertexts encrypted for that user
  - Honest users' privacy is not affected
  - Tracing operations can be delegated to clerks, running in parallel
- Users can claim their own ciphertexts and disclaim other ciphertexts

Our Contribution: precise modeling, construction in the standard model

Model of Traceable Group Encryption

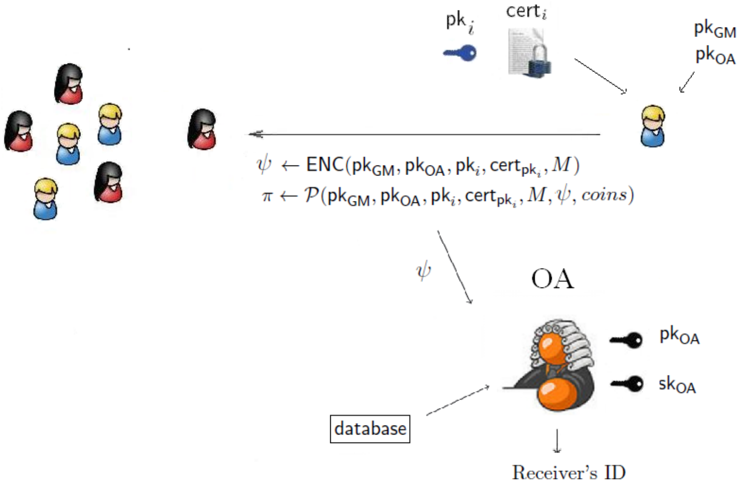
- Involve a non-interactive (*i.e.*, 2-round) join protocol



- Users generate their key pair on their own; no proof of knowledge of sk_i and no rewind in security proofs
- Made possible using structure-preserving signatures (Abe *et al.*, Crypto'10)

Model of Traceable Group Encryption

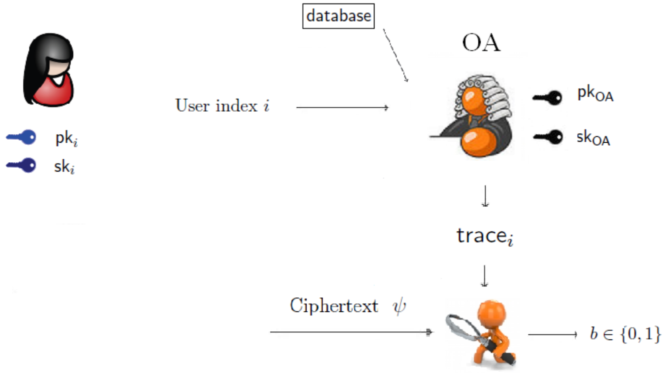
Group Encryption syntax



Model of Traceable Group Encryption

Additional functionalities of *Traceable* Group Encryption

- Implicit tracing mechanism:



- Claiming capability: using sk_i and a ciphertext ψ, user $\mathcal{U}_i$ can generate a claim / disclaimer τ

Security Model

- **Message security**: CCA2-security of honest receivers against colluding dishonest GM and OA
- **Anonymity** (a.k.a. key privacy): CCA2-anonymity of ciphertexts
  - Preserved against dishonest GM
  - Subsumes the CCA2-key privacy of the receiver's encryption scheme
  - . . . and the IND-CCA2 security of the OA's encryption scheme
- **Soundness**: no coalition of OA with dishonest groups members can
  - Produce a ciphertext ψ with a valid proof π such that $\mathit{Open}(\psi, \mathsf{sk}_{\mathrm{OA}}) = \perp$
  - Output a ciphertext-proof pair whose opening disagrees with the implicit tracing mechanism
- **Claiming Soundness**: users cannot disclaim their own ciphertexts or "hijack" other users' ciphertexts

Our Construction: Ingredients

- Assumes a common reference string (like [KTY07, CLY09,EAJ13])
- Uses Groth-Sahai proof systems (Eurocrypt'08) and the Linear assumption
- Uses structure-preserving signatures (Abe *et al.*, Crypto'10) as membership certificates
- . . . and CCA2-secure public key encryption schemes:
  - The Libert-Yung DLIN-based CCA2-secure cryptosystem (TCC'12): anonymity and built-in proofs of ciphertext validty
  - Kiltz's tag-based encryption scheme (publicly verifiable ciphertext validity)

Our Construction: Outline

- Users' keys are of the form

$$\mathsf{pk} = (X_1, X_2, \Gamma_1, \Gamma_2) = (g_1^{x_1} g^{x_0}, g_2^{x_2} g^{x_0}, g^{\gamma_1}, g^{\gamma_2}) \in \mathbb{G}^4$$

- GM holds a key pair $(sk_{\mathrm{GM}}, pk_{\mathrm{GM}})$ for a structure-preserving signature which allows certifying $\mathsf{pk} = (X_1, X_2, \Gamma_1, \Gamma_2)$

- During the Join protocol, user sends a *verifiable encryption* Φ_{venc} of $trace_i = g^{\gamma_1 \gamma_2}$ under pk_{OA}, where $(g, \Gamma_1, \Gamma_2, g^{\gamma_1 \gamma_2})$ is a Diffie-Hellman tuple

- Each TGE ciphertext carries a *traceability component*

$$(T_1, T_2, T_3) = \left(g^{\delta},\ \Gamma_1^{\delta/\omega},\ \Gamma_2^{\omega}\right)$$

  such that $trace_i = g^{\gamma_1 \gamma_2}$ solves the CDH instance (T_1, T_2, T_3)

- Ciphertext must include $T_4 = (\Lambda_0^{\mathsf{VK}} \cdot \Lambda_1)^{\delta}$, where $(\mathsf{SK}, \mathsf{VK})$ allows one-time signing the whole ciphertext

Our Construction: Outline

- Each TGE ciphertext contains a *traceability component*

$$(T_1, T_2, T_3) = \left(g^{\delta},\ \Gamma_1^{\delta/\omega},\ \Gamma_2^{\omega}\right)$$

such that $trace_i = g^{\gamma_1\gamma_2}$ allows testing $e(T_1, g^{\gamma_1\gamma_2}) = e(T_2, T_3)$

- Using $(\gamma_1, \gamma_2) \in \mathbb{Z}_p^2$, user can claim $(T_1, T_2, T_3) = \left(g^{\delta},\ \Gamma_1^{\delta/\omega},\ \Gamma_2^{\omega}\right)$ by computing $T_1^{\gamma_1} = \Gamma_1^{\delta}$ such that $e(T_1^{\gamma_1}, \Gamma_2) = e(T_2, T_3)$

  ...and proving knowledge of g^{1/γ_1} using a Groth-Sahai CRS "bound" to the ciphertext (cf. Malkin-Teranishi-Vahlis-Yung, TCC'11)

- Disclaiming proceeds similary

TGE Scheme for the Diffie-Hellman relation

A scheme for the Diffie-Hellman relation $\mathcal{R} = \{\big((X, Y), W\big) | e(g, W) = e(X, Y)\}$.

- Encryption phase:
  - Sender encrypts W under pk_i using a CCA2-anonymous encryption scheme
  - . . . and pk_i under $\mathsf{pk}_{\mathsf{OA}}$ using a CCA2-secure system
- Proof generation:
  - Compute commitments to pk_i and $\mathsf{cert}_{\mathsf{pk}_i}$
  - Prove that (i) commitments contain a valid pair $(\mathsf{pk}_i, \mathsf{cert}_{\mathsf{pk}_i})$; (ii) pk_i is the key encrypted under $\mathsf{pk}_{\mathsf{OA}}$; (iii) consistency with traceability components
  - Prove that W satisfies $\mathcal{R}$

Our Construction: Security

Relies on the hardness of the following problem:

- The q-**SFP Problem**: given $(g_z, h_z, g_r, h_r, a, \tilde{a}, b, \tilde{b}) \in \mathbb{G}^8$ and tuples $\{(z_j, r_j, s_j, t_j, u_j, v_j, w_j)\}_{j=1}^{q}$ s.t.

$$\begin{aligned} e(a, \tilde{a}) &= e(g_z, z_j) \cdot e(g_r, r_j) \cdot e(s_j, t_j) \\ e(b, \tilde{b}) &= e(h_z, z_j) \cdot e(h_r, u_j) \cdot e(v_j, w_j), \end{aligned}$$

  find a new such tuple $(z^\star, r^\star, s^\star, t^\star, u^\star, v^\star, w^\star)$ with $z^\star \neq 1_{\mathbb{G}}$

- The **Decision Linear** problem: given $(g, g_1, g_2, g_1^a, g_2^b, Z)$, decide if $Z = g^{a+b}$ or $Z \in_R \mathbb{G}$

- The **Decision** 3-**party Diffie-Hellman** assumption: given (g, g^a, g^b, g^c, η) decide if $\eta = g^{abc}$ or $\eta \in_R \mathbb{G}$

Summary

Contributions:

- Security model for *Traceable* Group Encryption
- Efficient non-interactive construction in the standard model

  Ciphertexts and proofs fit within 2.18kB and 9.38kB
  at the 128-bit security level

Open problems:

- Practical construction with shorter proofs
- Improve the efficiency for general pairing-product equation

Thanks!