Ciphertext-Policy Attribute-Based Broadcast Encryption with Small Keys

Benjamin Wesolowski (EPFL) Pascal Junod (HES-SO/HEIG-VD)

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

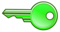
HAUTE ÉCOLE D'INGÉNIERIE ET DE GESTION DU CANTON DE VAUD
www.heig-vd.ch

This work was supported by the HES-SO and the EUREKA-Celtic+ H2B2VS project.

ICISC, Seoul, South Korea, November 25th, 2015

Plan

A Typical Pay-TV Scenario

«Grant access to all receivers having rights  AND . »

Attribute-Based Encryption

- In practice, one can often group decrypting entities by common properties, or **attributes**:
  - «receivers located in Seoul», «receivers located in a rural zone»,...
  - «receivers supporting SD», «receivers supporting HD», «receivers supporting 4K»,...
  - «receivers at patch level 3.2», «receivers at patch level 3.3»,...
- Idea of **attribute-based encryption (ABE)** proposed by Sahai and Waters (Eurocrypt'05) as a generalization of identity-based encryption.
- Roughly: give (individualized) attributes to receivers, and describe which receivers can decrypt a ciphertext with an **access equation** $\mathbb{A}$.

Ciphertext vs. Key Policy ABE

- **Ciphertext policy**: access policies are embedded into the ciphertext.
- **Key policy**: access policies are embedded into decryption keys.
- NB: in a Pay-TV scenario, access policies are rather dynamic (because of marketing guys), while changing decryption keys in a receiver is a very expensive operation.

Broadcast Encryption

- Concept introduced by Berkovits (Eurocrypt'91) and Fiat and Naor (Crypto'93)
- Idea: broadcast a ciphertext that only non-**revoked** receivers can decrypt.
- Collusion resistance: revoked receivers colluding together by **sharing their decryption key material** should not be able to decrypt a ciphertext as well.
- In the following of this talk:
  - Set of users (or receivers) is $\mathcal{U}$, with $n = |\mathcal{U}|$.
  - Set of revoked receivers is $\mathcal{R}$, with $\ell = |\mathcal{R}|$.
  - **Broadcast encryption scheme**: $n - \ell \ll n$.
  - **Revocation system:** $\ell \ll n$.

Attribute-Based Broadcast Encryption (ABBE)

- In some sense, an ABE is nothing but a BE: group of allowed receivers are defined by the access equation $\mathbb{A}$.
- Question: how can you **efficiently** revoke a (rogue) receiver?
- When using an ABE, dedicating a different attribute to each receivers is not efficient:
  - Public and private key size
  - Decryption time
  - Static nature of receivers
- Concept of **Attribute-Based Broadcast Encryption (ABBE)** proposed by Lubicz and Sirvent (Africacrypt'08)
  - ABE scheme with the additionnal functionality of revoking individual receivers in an efficient way.

Ciphertext-Policy ABBE

- $\mathsf{Setup}(\lambda) \to (\mathsf{pk}, \mathsf{msk})$: randomized algorithm which takes a security parameter λ as input and outputs a public key pk and a master key msk.
- $\mathsf{KeyGen}(u, \omega, \mathsf{msk}, \mathsf{pk}) \to \mathsf{dk}_u$: randomized algorithm that takes as input a receiver $u \in \mathcal{U}$, a set of attributes $\omega \subset \mathcal{B}$, msk and pk. It outputs a private decryption key $\mathsf{dk}_{(u,\omega)}$ for receiver u.
- $\mathsf{Encrypt}(\mathcal{R}, \mathbb{A}, \mathsf{pk}) \to (\mathsf{hdr}, \mathsf{k})$: randomized algorithm that takes as input a set of revoked receivers $\mathcal{R} \subset \mathcal{U}$, a Boolean access policy in CNF $\mathbb{A}$ and pk. It outputs a header hdr and a session key k.
- $\mathsf{Decrypt}(\mathsf{hdr}, (\mathcal{R}, \mathbb{A}), \mathsf{dk}_{(u,\omega)}, (u, \omega), \mathsf{pk}) \to \mathsf{k}$ or $\perp$: algorithm taking as input a header hdr, a set of revoked receivers $\mathcal{R}$, an access policy $\mathbb{A}$, a decryption key $\mathsf{dk}_{(u,\omega)}$ for receiver u equipped with attributes ω and pk. It outputs the session key k if and only if ω satisfies $\mathbb{A}$ and u is not in $\mathcal{R}$; otherwise, it outputs $\perp$.

Plan

Intuitive Description

- (Secure) combination of the Boneh-Gentry-Waters (Crypto'05) broadcast encryption scheme and of the Lewko-Sahai-Waters (IEEE Security & Privacy 2010) revocation system.
- Similar idea behind the Junod-Karlov (DRM'10) ABBE scheme.
- Boneh-Gentry-Waters has a PK and a DK size that depend on the number of entities in the system, and a constant-size ciphertext.
- Lewko-Sahai-Waters has a ciphertext linearly dependent on the number of revoked users, but the sizes of PK and DK are independent of the total number of users.

Setup

- Two groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order $p > 2^\lambda$ as well as a non-degenerate bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.
- Two non-zero elements $g, h = g^\xi \in \mathbb{G}$ and seven random exponents $\alpha, \gamma, b, \beta, \delta, r$ and r' in $\mathbb{Z}/p\mathbb{Z}$.
- We note $g_i = g^{\alpha^i}$.
- The public key pk consists of the elements of $\mathbb{G}$ g, $g_n^{\gamma r'}$, g^r, $g_{n+1}^{rr'}$, $g_{n+1}^{rr'b}, g_{n+1}^{rr'b^2}, h^{b\alpha^{n+1}r'r}, g^{\delta r}, g_n, \left(g^r_{\imath(a)}\right)_{a\in\mathcal{B}^*}$, and the two elements of $\mathbb{G}_T$ $e(g_1, g_n)^{rr'\beta\gamma}$ and $e(g_1, g_n)^{r\beta}$.

Key Generation

- Choose two random elements $\sigma_u, \varepsilon_u \in \mathbb{Z}/p\mathbb{Z}$.
- Define
  - $D_{u,0} = \left(g^{\gamma} g^{b^2 \sigma_u}\right)^{\varepsilon_u}$,
  - $D_{u,1} = \left(g^{b \cdot \mathrm{id}(u)} h\right)^{\sigma_u \varepsilon_u}$,
  - $D_{u,2} = g^{-\sigma_u \varepsilon_u}$,
  - $D_{u,3} = g_1^{r(\beta+\varepsilon_u)}$.
- The private key of receiver u is

$$\mathrm{dk}_u = \left((D_{u,k})_{k=0}^{3}, \left(g_{\imath(a)}^{\varepsilon_u}\right)_{a \in \mathcal{B}^*}, \left(g_{n+1+\imath(a)}^{\varepsilon_u}\right)_{a \in \mathcal{B}^*}, \left(g_{\imath(a)}^{\delta \varepsilon_u}\right)_{a \in \mathfrak{B}(u)} \right).$$

Encryption

- Access policy $\mathbb{A} = \beta_1 \wedge \ldots \wedge \beta_N$, with $\beta_i = \beta_{i,1} \vee \ldots \vee \beta_{i,M_i}$
- a revocation set $\mathcal{R} \subset \mathcal{U}$,
- $s_0, \ldots, s_N \in \mathbb{Z}/p\mathbb{Z}$ at random and one defines $s = \gamma r' s_0 + \sum_{i=1}^{N} s_i$.
- $C = g_n^s = \left(g_n^{\gamma \cdot r'}\right)^{s_0} g_n^{\left(\sum_{i=1}^{N} s_i\right)}$.
- For all $i = 1, \ldots, N$, one defines the elements $C_{i,0} = g^{r s_i}$ and

$$C_{i,1} = \left(g^{r\delta} \prod_{a \in \beta_i} g^r_{n+1-\imath(a)} \right)^{s_i},$$

as well as the corresponding N parts of the header $\mathrm{hdr}_i = (C_{i,0}, C_{i,1})$.

Encryption (2)

- $C_0 = g_{n+1}^{rr's_0}$, and for each $u \in \mathcal{R}$,

$$C_{u,1} = g_{n+1}^{rr'bs_u} \text{ and } C_{u,2} = \left(g^{b^2 \mathrm{id}(u)} h^b\right)^{\alpha^{n+1} rr' s_u}.$$

- Let $\mathsf{hdr}_0 = (C_0, (C_{u,1})_{u\in\mathcal{R}}, (C_{u,2})_{u\in\mathcal{R}})$ and $\mathsf{hdr} = (C, \mathsf{hdr}_0, \ldots, \mathsf{hdr}_N)$.
- The global session key k is given by

$$\mathsf{k} = e(g_1, g_n)^{r\beta s}.$$

Decryption (Overview)

- If $u \in \mathcal{R}$ or if there exists $i \in \{1, ..., N\}$, such that $\beta_i \cap \mathfrak{B}(u) = \emptyset$, return $\perp$.
- For $i = 1, ..., N$, choose one satisfying attribute $a \in \beta_i \cap \mathfrak{B}(u)$ and compute a $\mathsf{k}_i^{\varepsilon_u}$ value, as well as $\mathsf{k}_0^{\varepsilon_u}$ (see the paper for the complete formulas).
- $\mathsf{k} = \frac{e(D_{u,3}, C)}{\prod_{i=0}^{N} \mathsf{k}_i^{\varepsilon_u}} = e(g_1, g_n)^{r\beta s}$.

Plan

Selective Security Model

- **Setup.** The adversary chooses a distribution of attributes $\mathfrak{B} : \mathcal{U} \to \mathcal{P}(\mathcal{B})$, declares a set of revoked receivers $\mathcal{R}^* \subset \mathcal{U}$ and an access policy $\mathbb{A}^*$. The challenger runs the Setup algorithm and gives the public key pk to the adversary $\mathcal{A}$.
- **Query phase 1.** The adversary is allowed to (adaptively) issue queries to the challenger for private keys dk_u for receivers $u \in \mathcal{U}$ such that either $u \in \mathcal{R}^*$ or $\mathfrak{B}(u)$ does not satisfy the policy $\mathbb{A}^*$, *i.e.*, receivers not able to decrypt a ciphertext.
- **Challenge.** After having run the encryption algorithm $\mathsf{Encrypt}(\mathcal{R}^*, \mathbb{A}^*, \mathsf{pk})$, the challenger gets a header hdr and a session key k. Next, he draws a bit b uniformly at random, sets $\mathsf{k}_b = \mathsf{k}$ and picks k_{1-b} uniformly at random in the space of possible session keys. He finally gives the triple $(\mathsf{hdr}, \mathsf{k}_0, \mathsf{k}_1)$ to the adversary.

Selective Security Model (2)

- **Query phase 2.** The adversary is again allowed to (adaptively) issue queries for private keys dk_u for receivers $u \in \mathcal{U}$ such that either $u \in \mathcal{R}^*$ or $\mathfrak{B}(u)$ does not satisfy the policy $\mathbb{A}^*$.
- **Guess.** The adversary outputs a guess bit b'.

Proofs of Security

Definition (GDHE Decisional Problem, Boneh-Boyen-Goy, Crypto'05)

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two groups of prime order p, g a generator of $\mathbb{G}$, and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ a non-degenerate bilinear map. Let $f \in \mathbb{F}_p[X_1, ..., X_n]$ be a polynomial in n variables over $\mathbb{F}_p$, the finite field with p elements, and $P, Q \subset \mathbb{F}_p[X_1, ..., X_n]$ be two sets of polynomials, both containing 1. Choose $x_1, \ldots, x_n \in \mathbb{F}_p$ and $U \in \mathbb{G}_T$ uniformly at random. Given the elements

$$g^{\pi(x_1,\ldots,x_n)} \text{ and } e(g,g)^{\rho(x_1,\ldots,x_n)}$$

for each $\pi \in P$ and $\rho \in Q$, the *Generalized Diffie-Hellman Exponent (GDHE) Decisional Problem* is the problem of distinguishing $e(g,g)^{f(x_1,\ldots,x_n)}$ from U.

Security Proof (1)

Lemma

If the adversary $\mathcal{A}$ solves the CP-ABBE selective security game with advantage ε, then a simulator can be constructed to solve the (P, Q, f)-GDHE problem with advantage ε in polynomial time, with one oracle call to $\mathcal{A}$.

Security Proof (2)

Theorem

For any probabilistic algorithm $\mathcal{A}$ that totalizes at most q queries to the oracle performing group operations in $(\mathbb{G}, \mathbb{G}_T)$ and evaluations of $e(\cdot, \cdot)$, and declaring a set of revoked receivers of size at most η, as well as an access policy with at most N clauses ($\mathbb{A} = \beta_1 \wedge \cdots \wedge \beta_N$), then $\mathrm{Adv}^{\mathrm{ind}}(\lambda, \mathcal{U}, \mathcal{B}, \mathcal{A})$ is smaller or equal to

$$\frac{(q + 4(N + N + \eta) + 22 + |\mathcal{U}|(10N + 8))^2(8N + 3)}{2^{\lambda - 1}}.$$

Performances

Scheme	**Acc. Struct.**	pk size	dk_u size	hdr size
Attrapadung-Imai (2009)	Monotone	$O(N+n)$	$O(N+n)$	$O(\nu)$
Lubicz-Sirvent (2008)	AND & NOT	$O(N+n)$	$O(k_u)$	$O(\nu+\ell)$
Junod-Karlov (2010)	CNF	$O(N+n)$	$O(N+n)$	$O(\bar{\nu})$
Zhou-Huang (2010)	AND & NOT	$O(N+\log n)$	$O(N+\log n)$	$O(\log n)$
Li-Zhang (2015)	Monotone	$O(N+n)$	$O(k_u+n)$	$O(\nu)$
This paper	CNF	$O(N)$	$O(N)$	$O(\bar{\nu}+\ell)$

Legend: k_u is the number of attributes assigned to a receiver $u \in \mathcal{U}$, ν the length of the access structure, $\bar{\nu}$ the number of clauses in a CNF access structure, $N = |\mathcal{B}|$, $n = |\mathcal{U}|$ and $\ell = |\mathcal{R}|$.

Plan

Practical Implementation

- Scheme implemented in C++ with Stanford's open-source *Pairing-Based Cryptography (PBC)* library.
- Group with a 160-bit order and a 512-bit base field order.
- Scenario with 5 attributes run on an Intel Core i7 clocked at 2.3 GHz.
  - Setup phase (including public key generation): 237 ms
  - Private key generation (for each receiver): 75 ms
  - Decryption of a message with 3 clauses without revocation: 25 ms
  - Each revocation adds 4 ms to the decryption time.

Thank you

고맙습니다

The full version of this paper is available at
http://eprint.iacr.org/2015/836.