Attribute-based Encryption Resilient to Auxiliary Input

Zhiwei Wang

Siu Ming Yiu

Outline

* Problem Statement
* Attribute-Based Encryption with Auxiliary
* Our Techniques

Side-Channel Attack

* The central notion of modern cryptography relies on the secrecy of the secret key.
* In practice, this paradigm is subject to the immanent threat of side-channel attacks.

Leakage-Resilient Cryptography

* Formal security guarantees even when the secret (key/randomness) leaks
* Here we only consider memory leakage.
* The adversary is allowed to specify an efficiently computable leakage function f
  * Obtain the output of f applied to the secret
  * Aims to model the possible leakage in practice

A Major Open Problem

* [Goldwasser @ Eurocrypt '09 Invited Talk]
* *allowing for continuous unbounded leakage*
* *without additionally restricting its type*

* [AGV09, NS09, ADNSWW10, BKKV10, CDRW10, DGKPV10, DHLW10, LLW11, LRW11…]

Bounded Retrieval Model

* Allowed bits of leakage is *l*
* *l* is also a system parameter
* Size of the secret key increases with *l*
* But *l* does not affect public key size, communication and computation efficiency
* e.g., [ADNSWW10, CDRW10]
* Hope the attack is detected and stopped before the whole secret is leaked

Auxiliary Inputs

* Any *f* that no poly. time adversary can invert
* E.g., One-way permutation (OWP)
* OWP is not allowed in the relative model
* [DGKPV10] proposed public-key encryption (PKE) schemes with auxiliary inputs
* [YSY12] proposed ABE schemes with auxiliary inputs
* All these bound the leakage throughout the *entire lifetime* of the secret key

Continual Leakage Model

* Allows for continuous memory leakage (CML)
* Continually updates / refreshes the secret key
* Leakage between updates are still bounded
* [DHLW10]: signature and identification
* [BKKV10]: signature, PKE, and selective-ID IBE
* [LLW11]: signature and PKE
* [Zhang13]: ABE

ABE with Auxiliary Inputs

* ABE found many applications
* Resilience => composition of Attribute-based systems
* A “clean” security definition
  * Free from numeric bounds

Continual-Leakage-Resilient ABE

* Current CML models for ABE consider leakage of the current secret key for a given time only
  * [Zhang13]
* The old secret key should be *securely* erased.
* Less disastrous leakage => Less benefits

Problem Statement

- We tackle the problem of "*allowing ABE for continuous unbounded leakage, without additionally restricting the type of leakage*".
- [DGKPV10]: PKE, no continual leakage
- [BKKV10]: IBE,selective-ID, no leakage from *msk*
- [LRW11]: IBE, adaptive-ID, leakage size bounded
- [YSY12]: IBE, adaptive-ID

Our Contributions

* We propose the first CP-ABE scheme that is secure in the presence of auxiliary inputs
  * Adaptive security in the Standard Model
  * Based on Static Assumptions
  * Moderate costs (ctxt. size, comp. complexity)
* We propose the first KP-ABE scheme resilience to auxiliary inputs
* We impove our ABE schemes secure in the presence of continual auxiliary model

Goldreich-Levin Theorem

* The key technique in [DGKPV10] is the modified Goldreich-Levin (GL) theorem.
* The original GL theorem is over $GF(2)$
  * For an uninvertible function h: $GF(2)^m \rightarrow \{0, 1\}^*$,
  * $\langle \boldsymbol{e}, \boldsymbol{y} \rangle \in GF(2)$ is pseudorandom
  * given $h(\boldsymbol{e})$ and uniformly random $\boldsymbol{y}$

Modified GL Theorem

* Let q be a prime
* H be a poly(m)-sized subset of GF (q)
* $h : H^m \rightarrow \{0,1\}^*$ be any (randomized) function
* If there is a PPT algorithm D that distinguishes between $\langle \boldsymbol{e}, \boldsymbol{y} \rangle$ and the uniform distribution over $GF(q)$ given $h(\boldsymbol{e})$ and $\boldsymbol{y} \leftarrow GF(q)^m$
* then there is a PPT algorithm A that inverts h with probability $1/(q^2 \cdot \text{poly}(m))$

Aux-PKE -> Aux-ABE

* Attribute-based secret key has “structure”
  * Not a λ-bit number
  * Secret random factors from a small domain
  * The size of attribute-based secret key is according to the number of attributes

Aux-PKE + LR-ABE -> Aux-ABE?

* Even worse, many many secret keys in ABE…
* Leak "semi-functional" (SF) keys in simulation
* SF-key is perturbed from a real key by *m* blinding factors from $\mathbf{Z}_p$ where p is of size 2^{λ}.
* Inefficient invertor if we followed
* Countermeasure for leakage just appears in the security proof but not the actual scheme.

Our Auxiliary Input Model

- Usual secure against chosen-plaintext attack (CPA)
- Leakage oracle (LO) in additional to Key Extraction oracle (KEO)
- LO takes an input of $f \in$ **F** and S returns $f(msk, sk_S, mpk, S)$
- No LO query after challenge phase
- **F**: Given *mpk*, S*, $\{f_i(msk, sk_{Si}, mpk, S_i)\}$, and a set of secret keys w/o sk_{Si}, no PPT algo. can output a secret key sk_{S*} of S*

Here are the parameters, I will keep *msk* from you

I want $f0(msk)$, $f1(sk_{S1})$, sk_{S4}, sk_{S1} and $f3(msk, sk_{S4})$

Sure, just make your adaptive choices

I want to be challenged with these 2 messages: m_0, m_1

Now I encrypt a random 1 of them, make your guess

Roadmap of Our Construction

Lewko-Waters Adaptive-ID IBE

Lewko-Rouselakis-Waters LR-IBE

Yuen-Chow-Zhang-Yiu IBE with Auxiliary Inputs

Zhang-Shi-Wang-Chen-Mu LR-ABE

Our ABE with Auxiliary Inputs

Leakage via Dual System

* We know how to “fake” everything!
* We can leak them too.
* Caution: leaking can’t spoil faking.
* Correlation regarding SF objects is information-theoretically (IT) hidden

Our Design Constraints

* Small blinding factors are used in SF key
* When the key is leaked, uninvertible function of key can be created from uninv.-func. of factors
* Inner product = 0 => Exponent in $\mathbf{G}_q$ = 0
* Use modified GL theorem to ensure the indistinguishability of 2 types of SF keys.

Our Contributions (2)

* For the security poof, we propose three improved statics assumptions, and prove them in appendix.

Function Family

- Basic: Given *mpk*, S*, $\{f_i(msk, sk_{Si}, mpk, S_i)\}$, and a set of secret keys w/o sk_{Si}, no PPT algo. can output a secret key sk_{S*} of S*
- CAL: Given *mpk*, S*, $\{f_i(L_{msk}, L_S, msk, sk_{Si}, mpk, S_i)\}$, and a set of secret keys w/o *any valid* sk_{Si}, no PPT algo. can output sk_{S*} of S*
- The lists *L*'s include all keys ever produced
- Additionally, may give leakage during setup

* Thank! Any questions?