无密文扩展的CPA到CCA转换：适用于常量密文大小的门限属性基加密

主讲人：丁昀　　主讲时间：5.25

Information Sciences 613 (2022) 966–981

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Generic conversions from CPA to CCA without ciphertext expansion for threshold ABE with constant-size ciphertexts

Jianchang Lai [a,*], Fuchun Guo [b,*], Willy Susilo [b], Peng Jiang [c], Guomin Yang [d], Xinyi Huang [e]

[a] School of Cyber Science and Engineering, Southeast University, Nanjing, China
[b] Institute of Cybersecurity and Cryptology, School of Computing and Information Technology, University of Wollongong, Australia
[c] School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China
[d] School of Computing and Information Systems, Singapore Management University, Singapore
[e] The Hong Kong University of Science and Technology(Guangzhou), Guangzhou, China

ARTICLE INFO

Article history:
Received 27 November 2020
Received in revised form 12 August 2022
Accepted 24 August 2022
Available online 6 September 2022

ABSTRACT

CCA security is desirable when designing encryption schemes because it captures active attackers. One efficient approach for achieving CCA security is to use generic conversions. We first design a CPA secure scheme, and then transform it into a CCA secure scheme using the Fujisaki-Okamoto technique (CRYPTO '99) in the random oracle model or the Canetti-Halevi-Katz technique (EUROCRYPT '04) in the standard model if the CPA secure scheme

目录

01 研究背景
02 背景知识
03 核心算法
04 实验与结果

01

01 研究背景

属性基加密（ABE）

选择明文攻击
Chosen Plaintext Attack

敌手能够访问加密预言机（即加密服务），可以自由选择任意明文并获取对应的密文，但无法直接获取密钥或解密其他密文。

目标是利用这些信息推断密钥或破坏加密方案的不可区分性

03

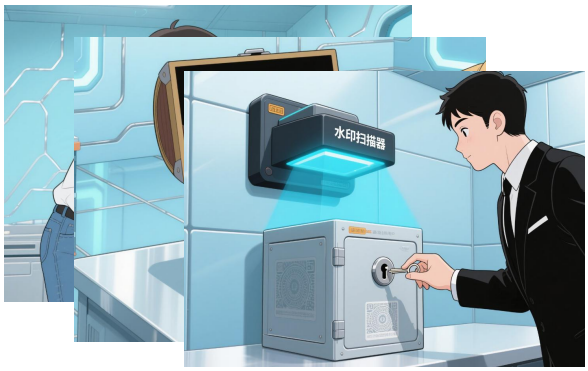
03 核心算法

1. Setup（系统初始化）
   - **输入**：安全参数λ、通用属性集$\mathscr{P}$、整数n。
   - **操作**：运行原方案Π的$\text{Setup}(\lambda, \mathscr{P} \cup W)$生成主密钥对$(mpk, msk)$，选择哈希函数$H : \{0,1\}^* \to \{0,1\}^n$（安全证明中视为随机预言机），最终$mpk = (mpk, n, H)$，$msk$沿用原方案结果。通过引入虚拟属性集$W$，扩展属性空间，为抵御主动攻击奠定基础。
2. KeyGen（私钥生成）
   - **输入**：主密钥对(mpk, msk)、属性集$A \subset \mathscr{P}$。
   - **操作**：设置$A_e = A$，直接调用原方案的$\text{KeyGen}(mpk, msk, A_e)$生成私钥$sk_{A_e}$，返回$sk'_A = sk_{A_e}$，保持私钥生成逻辑，确保与原方案兼容性。
3. Encrypt（加密）
   - **输入**：mpk、消息M、属性集S、阈值t（$1 \le t \le |S|$）。
   - **步骤**：
     - 随机选取r，调用原方案$\text{Enc}_M(mpk, r, M)$生成CT_1。
     - 计算$H(CT_1, S, t) = c_1 c_2 \cdots c_n$，定义虚拟属性集$I_C = B_{c_1,1} \cup B_{c_2,2} \cup \cdots \cup B_{c_n,n}$。
     - 扩展策略为$(S_e, t_e) = (S \cup I_C, t)$，调用$\text{Enc}_P(mpk, r, S_e, t_e)$生成$CT_2$。
     - 输出密文$CT_e = (CT_1, CT_2)$。通过哈希动态关联虚拟属性，隐藏真实策略，抵御密文篡改。
4. Decrypt（解密）
   - **输入**：mpk、密文CT_e、属性集A（$|A \cap S| \ge t$）。
   - **步骤**：
     - 生成扩展策略(S_e, t_e)（同加密逻辑）。
     - 运行$\text{Verify}(mpk, CT_e, S_e, t_e)$，若输出非1，返回$\perp$（拒绝无效密文）。
     - 调用原方案$\text{Decrypt}(mpk, CT_e, (S_e, t_e), sk'_A)$解密，返回结果。利用可验证性确保密文合法性，再执行解密。

4.1. Generic Construction I: from Verifiability

4.2. Generic Construction II: from Delegatability

1. Setup（系统初始化）
   - **输入**：安全参数λ、通用属性集$\mathscr{P}$、整数n。
   - **操作**：
     - 运行原方案Π的$\text{Setup}(\lambda, \mathscr{P} \cup W)$生成主密钥对$(mpk, msk)$。
     - 选择哈希函数$H : \{0,1\}^* \to \{0,1\}^n$（在随机预言机模型中，$H$由挑战者控制）。
     - 最终输出$mpk = (mpk, n, H)$，msk沿用原方案结果。通过引入虚拟属性集W，扩展系统属性空间，为抵御主动攻击构建基础框架。
2. KeyGen（私钥生成）
   - **输入**：主密钥对(mpk, msk)、属性集$A \subset \mathscr{P}$。
   - **操作**：
     - 利用可委托性，将属性集扩展为$A_e = A \cup W$。
     - 调用原方案的$\text{KeyGen}(mpk, msk, A_e)$生成私钥$sk_{A_e}$，返回$sk'_A = sk_{A_e}$。通过扩展属性集，结合可委托性，确保私钥生成既兼容原方案又满足新安全模型需求。
3. Encrypt（加密）
   - **输入**：mpk、消息M、属性集S、阈值t（$1 \le t \le |S|$）。
   - **步骤**：
     - 随机选取r，调用原方案的$\text{Enc}_M(mpk, r, M)$生成CT_1。
     - 计算$H(CT_1, S, t) = c_1 c_2 \cdots c_n$，定义虚拟属性集$I_C = B_{c_1,1} \cup B_{c_2,2} \cup \cdots \cup B_{c_n,n}$。
     - 扩展策略为$(S_e, t_e) = (S \cup I_C, t + |I_C|)$，调用$\text{Enc}_P(mpk, r, S_e, t_e)$生成$CT_2$。
     - 输出密文$CT_e = (CT_1, CT_2)$。通过哈希动态关联虚拟属性，隐藏真实策略，同时利用策略扩展增强抗攻击能力。
4. Decrypt（解密）
   - **输入**：mpk、密文CT_e、属性集A（$|A \cap S| \ge t$）。
   - **步骤**：
     - 生成扩展策略$(S_e, t_e) = (S \cup I_C, t + |I_C|)$。
     - 利用可委托性，运行$\text{Delegate}(mpk, msk, sk'_A, A \cup W \cup I_C)$生成委托私钥$sk_{A \cup W \cup I_C}$。
     - 调用原方案的$\text{Decrypt}(mpk, CT_e, (S_e, t_e), sk_{A \cup W \cup I_C})$解密，返回结果。通过可委托性生成符合扩展策略的私钥，确保解密流程在新安全模型下的有效性。



攻击方式	虚拟属性作用	Verifiability / Delegatability 作用
攻击者伪造密文	没有虚拟属性 → 解密失败	/
攻击者做密钥查询	/	Delegatability：模拟器派生子密钥
攻击者做解密查询	验证虚拟属性，防伪伪密文	Verifiability：用模拟密钥解出正确结果

两种算法流程看似接近，核心差别在于：

- **Construction I：强调解密结果一致性**
- **Construction II：强调密钥授权与派生**

	Construction I	Construction II
场景	多人用钥匙开同信	老板授权助理开信
关键验证	内容+水印必须一致	授权派生钥匙+水印检验
防伪方式 核心价值	防伪水印验证 保证多钥匙解密一致	防伪水印验证 保证授权密钥安全解密

	Construction I	Construction II
依赖性质	Verifiability	Delegatability
解密前操作	解密→验证虚拟属性	解密→验证虚拟属性
安全性证明	多密钥解密结果一致，便于模拟解密	派生密钥解密，便于模拟密钥查询
难点控制	模拟解密一致性	模拟密钥派生一致性

Construction I：像是多钥匙多锁盲，一致才算真信
Construction II：像是老板授权助理开信，确保助理开的信和老板开的一样，而且防伪水印检验过关