可证明安全 - 密码学中的安全性证明

钱宸
网络空间安全学院
山东大学

2025.09.15

Contents

基于规约的证明

什么是安全性规约?

- 安全性规约 (Security Reduction)
- 规约: 将一个复杂问题的安全性转化为一个" 已知" 问题的安全性
- 例如: 将一个加密方案的安全性规约到一个数学难题 (如大整数分解问题) 的难解性

什么是安全性规约?

- 一个密码方案 Π 和一个数学难题 P
- 假设已知 P 在多项式时间内是难解的
- 目标: 密码方案的安全性在多项式时间内是难解的

什么是安全性规约?

从复杂度角度出发 - NP 困难问题

- **NP 完全问题 A:** 3SAT, 哈密顿回路, 图三着色等等.

什么是安全性规约?

从复杂度角度出发 - NP 困难问题

- **NP 完全问题 A:** 3SAT, 哈密顿回路, 图三着色等等.
- **NP 困难问题 B:** " 比 **NP** 完全问题更难的问题".

什么是安全性规约?

从复杂度角度出发 - NP 困难问题

- **NP 完全问题 A:** 3SAT, 哈密顿回路, 图三着色等等.
- **NP 困难问题 B:** " 比 **NP** 完全问题更难的问题".
- **B 比 A 更难:** “如果存在一个多项式时间算法 P_B 能够解决 B，那么也存在一个多项式时间算法 P_A 能够解决 A”

什么是安全性规约?

从复杂度角度出发 - NP 困难问题

- **NP 完全问题 A:** 3SAT, 哈密顿回路, 图三着色等等.
- **NP 困难问题 B:** " 比 **NP** 完全问题更难的问题".
- **B 比 A 更难:** “如果存在一个多项式时间算法 P_B 能够解决 B，那么也存在一个多项式时间算法 P_A 能够解决 A”
- 构造算法 (证明): $R(P_B) = P_A$

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.
- **Q 比 P 更难:** " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.
- **Q 比 P 更难:** " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"
- **规约 (证明):** $R(P_Q) = P_P$

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.
- **Q 比 P 更难:** " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"
- **规约 (证明):** $R(P_Q) = P_P$

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.
- **Q 比 P 更难:** " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"
- **规约 (证明):** $R(P_Q) = P_P$

规约合理性

- " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"

什么是安全性规约?

安全性说明

- **困难问题 P:** 大整数分解问题、离散对数、格最短向量等等.
- **加密算法的安全性 Q:** 根据密文找出原文、根据公钥找出私钥等等.
- **Q 比 P 更难:** " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"
- **规约 (证明):** $R(P_Q) = P_P$

规约合理性

- " 如果存在一个多项式时间算法 P_Q 能够解决 Q, 那么也存在一个多项式时间算法 P_P 能够解决 P"
- **逆否命题:** " 如果不存在一个多项式时间算法 P_P 能够解决 P, 那么也不存在一个多项式时间算法 P_Q 能够解决 Q"

什么是安全性规约 - RSA 加密

例子: RSA 加密 [Rivest et al., 1978]

- RSA 加密的安全性规约到大整数分解问题
- " 存在一个多项式时间算法能够破解 RSA 加密" $\Longrightarrow$
  " 存在一个多项式时间算法来解决大整数分解问题"
- 通过规约, 我们可以将 RSA 的安全性建立在大整数分解问题的难解性上

规约的威力

我们看到规约在复杂度以及安全性论证中的重要性. 其实有更多的地方也在使用规约的思想, 例如:

- **困难性佐证:** 困难问题之间的安全性
- **算法设计:** 通过规约将构造一个复杂问题的算法化简成为另一个简单问题的算法

下面我们分别用一些例子来说明规约的威力.

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

目标

- 构造一个新的加密算法 $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$
- **安全性:** 根据密文找不到原文

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:
  1. $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^\lambda)$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:
  1. $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^\lambda)$
  2. $\mathsf{m} \hookleftarrow \mathsf{U}(\mathcal{M})$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:
  1. $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^\lambda)$
  2. $\mathsf{m} \hookleftarrow \mathsf{U}(\mathcal{M})$
  3. $\mathsf{ct} \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:
  1. $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^\lambda)$
  2. $\mathsf{m} \hookleftarrow \mathsf{U}(\mathcal{M})$
  3. $\mathsf{ct} \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$
  4. $\mathsf{m}' := \mathcal{A}(\mathsf{pk}, \mathsf{ct})$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

首先明确我们需要的安全性:

选择明文还原性 MR-CPA

- **攻击目的 MR- (Message Recovery):** 根据密文找出原文
- **攻击方式 CPA (Chosen Plaintext Attack):** 攻击者可以选择任意明文进行加密, 并获得对应的密文 (公钥加密的通常形式)
- **形式化的安全性定义 (安全性游戏):**
  $\mathbf{Exp}^{\text{MR-CPA}}_{\Pi,\mathcal{A}}(1^\lambda)$:
  1. $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KGen}(1^\lambda)$
  2. $\mathsf{m} \hookleftarrow \mathsf{U}(\mathcal{M})$
  3. $\mathsf{ct} \xleftarrow{\$} \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$
  4. $\mathsf{m}' := \mathcal{A}(\mathsf{pk}, \mathsf{ct})$
  5. **return** $[\![\mathsf{m} = \mathsf{m}']\!]$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

可用的工具: 计算性 Diffie-Hellman 假设

计算性 Diffie-Hellman 假设 [Diffie and Hellman, 1976]

- 给定 (g, g^a, g^b), 计算 g^{ab} 是困难的.
- 形式化定义: 对于所有多项式时间算法 $\mathcal{A}$, 有

$$\mathsf{Adv}^{\mathsf{CDH}}_{\mathbb{G},\mathcal{A}}(1^\lambda) = \Pr\left[\mathcal{A}(g, g^a, g^b) = g^{ab}\right] \leq \mathsf{negl}(\lambda).$$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

一些对比和联想

- **安全性需求:** 不能 “根据公钥和密文找出原文”
- **计算性 Diffie-Hellman 假设:** 不能 “给定 (g, g^a, g^b), 计算 g^{ab}”
- **简单类比:** 公钥 $\mathsf{pk} = (g, g^a)$, 密文 $\mathsf{ct} = (g^b, g^{ab} \cdot \mathsf{m})$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

El-Gamal 加密算法 [ElGamal, 1985]

- 公钥: (p, g, h), 私钥: x, 其中 $h = g^x \mod p$
- 加密: 对消息 m, 选择随机数 y, 计算密文 $(c_1, c_2) = (g^y \mod p, m \cdot h^y \mod p)$
- 解密: 使用私钥 x 计算 $m = c_2 / c_1^x \mod p$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}
- $\mathcal{B}$ 将 (p, g, g^a) 作为公钥, 并选择随机密文元素 $Z \in \mathbb{G}$, 构造密文 (g^b, Z)

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}
- $\mathcal{B}$ 将 (p, g, g^a) 作为公钥, 并选择随机密文元素 $Z \in \mathbb{G}$, 构造密文 (g^b, Z)
- $\mathcal{B}$ 将公钥和密文发送给 $\mathcal{A}$, 并获得 m'

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}
- $\mathcal{B}$ 将 (p, g, g^a) 作为公钥, 并选择随机密文元素 $Z \in \mathbb{G}$, 构造密文 (g^b, Z)
- $\mathcal{B}$ 将公钥和密文发送给 $\mathcal{A}$, 并获得 m'
- $\mathcal{B}$ 计算 $g^{ab} = Z/\mathsf{m}'$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}
- $\mathcal{B}$ 将 (p, g, g^a) 作为公钥, 并选择随机密文元素 $Z \in \mathbb{G}$, 构造密文 (g^b, Z)
- $\mathcal{B}$ 将公钥和密文发送给 $\mathcal{A}$, 并获得 m'
- $\mathcal{B}$ 计算 $g^{ab} = Z/\mathsf{m}'$

假设 $\Rightarrow$ 算法: 重新构建 El-Gamal 加密算法

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够在 MR-CPA 攻击模型下破解 El-Gamal 加密
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决计算性 Diffie-Hellman 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b), 目标是计算 g^{ab}
- $\mathcal{B}$ 将 (p, g, g^a) 作为公钥, 并选择随机密文元素 $Z \in \mathbb{G}$, 构造密文 (g^b, Z)
- $\mathcal{B}$ 将公钥和密文发送给 $\mathcal{A}$, 并获得 m'
- $\mathcal{B}$ 计算 $g^{ab} = Z/\mathsf{m}'$

定理 (El-Gamal 加密的 MR-CPA 安全性)

如果计算性 *Diffie-Hellman* 假设成立, 那么 *El-Gamal* 加密在 MR-CPA 攻击模型下是安全的.

假设 ⇒ 假设: CDH ⇒ DDH

计算性 Diffie-Hellman 假设 [Diffie and Hellman, 1976]

- 给定 (g, g^a, g^b), 计算 g^{ab} 是困难的.
- 形式化定义: 对于所有多项式时间算法 $\mathcal{A}$, 有

$$\mathsf{Adv}^{\mathsf{CDH}}_{\mathbb{G},\mathcal{A}}(1^\lambda) = \Pr\left[\mathcal{A}(g, g^a, g^b) = g^{ab}\right] \leq \mathsf{negl}(\lambda).$$

判定性 Diffie-Hellman 假设 [Diffie and Hellman, 1976]

- 给定 (g, g^a, g^b), 计算 g^{ab} 是困难的.
- 形式化定义: 对于所有多项式时间算法 $\mathcal{A}$, 对于任意 $c \in \mathbb{Z}_p$ 有

$$\mathsf{Adv}^{\mathsf{DDH}}_{\mathbb{G},\mathcal{A}}(1^\lambda) = \left|\Pr\left[\mathcal{A}(g, g^a, g^b, g^{ab}) = 1\right] - \Pr\left[\mathcal{A}(g, g^a, g^b, g^c) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)，目标是判定 Z 是否等于 g^{ab}

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)，目标是判定 Z 是否等于 g^{ab}
- $\mathcal{B}$ 使用 $\mathcal{A}$ 计算 $W = \mathcal{A}(p, g, g^a, g^b)$

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)，目标是判定 Z 是否等于 g^{ab}
- $\mathcal{B}$ 使用 $\mathcal{A}$ 计算 $W = \mathcal{A}(p, g, g^a, g^b)$
- $\mathcal{B}$ 比较 W 和 Z，如果相等则输出 1，否则输出 0

假设 $\Rightarrow$ 假设：DDH $\Rightarrow$ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)，目标是判定 Z 是否等于 g^{ab}
- $\mathcal{B}$ 使用 $\mathcal{A}$ 计算 $W = \mathcal{A}(p, g, g^a, g^b)$
- $\mathcal{B}$ 比较 W 和 Z，如果相等则输出 1，否则输出 0

假设 ⇒ 假设: DDH ⇒ CDH

安全性规约

- 假设存在一个多项式时间算法 $\mathcal{A}$ 能够解决 CDH 问题
- 构造一个多项式时间算法 $\mathcal{B}$ 来解决 DDH 问题
- $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z), 目标是判定 Z 是否等于 g^{ab}
- $\mathcal{B}$ 使用 $\mathcal{A}$ 计算 $W = \mathcal{A}(p, g, g^a, g^b)$
- $\mathcal{B}$ 比较 W 和 Z, 如果相等则输出 1, 否则输出 0

定理 (DDH ⇒ CDH)

如果判定性 *Diffie-Hellman* 假设成立, 那么计算性 *Diffie-Hellman* 假设也成立.

显然, 但是没什么用的引理: 如果判定性 Diffie-Hellman 假设成立, 那么 El-Gamal 加密在 MR-CPA 攻击模型下是安全的.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

多个 DDH 实例的安全性

给定敌手多个 DDH 实例是否能提高其判定成功率?

定义 (k-DDH 假设)

- $\mathcal{D}_0 = \left\{(g, g^{a_i}, g^{b_i}, g^{a_i b_i}) \mid \{a_i, b_i\}_{i\in\{1,\ldots,k\}} \hookleftarrow \mathsf{U}(\mathbb{Z}_p)\right\}$
- $\mathcal{D}_1 = \left\{(g, g^{a_i}, g^{b_i}, g^{c_i}) \mid \{a_i, b_i, c_i\}_{i\in\{1,\ldots,k\}} \hookleftarrow \mathsf{U}(\mathbb{Z}_p)\right\}$
- 对于所有 PPT(概率多项式图灵机) 敌手 $\mathcal{A}$, 有

$$\mathsf{Adv}^{k\text{-}\mathsf{DDH}}_{\mathbb{G},\mathcal{A}}(1^\lambda) = \left|\Pr[\mathcal{A}(\mathcal{D}_0) = 1] - \Pr[\mathcal{A}(\mathcal{D}_1) = 1]\right|.$$

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (k-DDH $\Rightarrow$ DDH)

如果对于任意整数 k, k-DDH 假设成立, 那么 DDH 假设也成立.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (k-DDH $\Rightarrow$ DDH)

如果对于任意整数 k, k-DDH 假设成立, 那么 DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 DDH 假设, $R(\mathcal{A})$ 能够成功攻击 k-DDH 假设.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (k-DDH $\Rightarrow$ DDH)

如果对于任意整数 k, k-DDH 假设成立, 那么 DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 DDH 假设, $R(\mathcal{A})$ 能够成功攻击 k-DDH 假设.

证明思路.

- 构造一个多项式时间算法 $\mathcal{B} = R(\mathcal{A})$ 来解决 k-DDH 问题
- $\mathcal{B}$ 的输入是 $(p, g, \left\{g^{a_i}, g^{b_i}, Z_i\right\})$, 目标是判定是否所有 Z_i 等于 $g^{a_i b_i}$.
- $\mathcal{B}$ 将 $(p, g, g^{a_1}, g^{b_1}, Z_1)$ 作为输入给 $\mathcal{A}$.
- $\mathcal{B}$ 使用 $\mathcal{A}$ 的输出作为自己的输出.

□

假设 $\Rightarrow$ 假设 2：DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立，那么对于任意整数 k，k-DDH 假设也成立.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造?**

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- 能否和上面一样进行构造?
- 不行!

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造?**
- **不行!**
- **Try:**

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造?**
- **不行!**
- **Try:**
  - $R(\mathcal{B}) = \mathcal{A}$

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造?**
- **不行!**
- **Try:**
  - $R(\mathcal{B}) = \mathcal{A}$
  - 想调用 $\mathcal{A}$, 但是 $\mathcal{A}$ 的输入是 $(p, g, \{g^{a_i}, g^{b_i}, Z_i\})$

假设 $\Rightarrow$ 假设 2：DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立，那么对于任意整数 k，k-DDH 假设也成立.

需要构造： R : PPT $\to$ PPT，使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设，$R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造？**
- **不行！**
- **Try:**
  - $R(\mathcal{B}) = \mathcal{A}$
  - 想调用 $\mathcal{A}$，但是 $\mathcal{A}$ 的输入是 $(p, g, \{g^{a_i}, g^{b_i}, Z_i\})$
  - $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

定理 (DDH 的随机自规约性)

如果 DDH 假设成立, 那么对于任意整数 k, k-DDH 假设也成立.

需要构造: R : PPT $\to$ PPT, 使得对于任意 PPT 敌手 $\mathcal{A}$ 能够成功攻击 k-DDH 假设, $R(\mathcal{A})$ 能够成功攻击 DDH 假设.

- **能否和上面一样进行构造?**
- **不行!**
- **Try:**
  - $R(\mathcal{B}) = \mathcal{A}$
  - 想调用 $\mathcal{A}$, 但是 $\mathcal{A}$ 的输入是 $(p, g, \{g^{a_i}, g^{b_i}, Z_i\})$
  - $\mathcal{B}$ 的输入是 (p, g, g^a, g^b, Z)
  - **需要将 $\mathcal{B}$ 的输入“扩展”成 $\mathcal{A}$ 的输入**

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

假如我们将群 $(\mathbb{G}, \cdot)$ 同构到 $(\mathbb{Z}_p, +)$ 的群上, 记 $\left[\cdot\right] : \mathbb{Z}_p \to \mathbb{G}$ 为该同构映射. 即 $\left[x\right] := g^x$, 其中 $g \in \mathbb{G}$ 为群 $\mathbb{G}$ 的生成元. 该记号可以推广到矩阵上

从矩阵的角度理解 DDH 假设

矩阵形式 DDH 假设: 给定矩阵 $A = \left[1 \quad a\right]^\mathsf{T}$, 其中 $a \leftarrow \mathsf{U}(\mathbb{Z}_p)$, 随机生成 $b, c \leftarrow \mathsf{U}(\mathbb{Z}_p)$, 则有下列计算不可区分性 (对于任意 PPT 敌手无法区分下列分布):

$$\begin{bmatrix} 1 \\ a \end{bmatrix} \cdot \begin{pmatrix} 1 & b \end{pmatrix} = \begin{bmatrix} 1 & b \\ a & ab \end{bmatrix} \approx_c \begin{bmatrix} 1 & b \\ a & c \end{bmatrix}$$

假设 $\Rightarrow$ 假设 2：DDH 的随机自规约性

如何将一个 DDH 实例“扩展”成多个 DDH 实例?

DDH 的随机自规约

- 给定一个实例 $\mathbf{D} = \begin{bmatrix} 1 & b \\ a & c \end{bmatrix}$,
- 计算 $\mathbf{E}_i = \begin{pmatrix} 1 & 0 \\ r_i & s_i \end{pmatrix} \cdot \mathbf{D} \cdot \begin{pmatrix} 1 & t_i \\ 0 & 1 \end{pmatrix}$, 其中 $r_i, s_i, t_i \leftarrow\!\!\shortmid \mathsf{U}(\mathbb{Z}_p)$.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

如何将一个 DDH 实例“扩展”成多个 DDH 实例?

DDH 的随机自规约

- 给定一个实例 $\mathbf{D} = \begin{bmatrix} 1 & b \\ a & c \end{bmatrix}$,
- 计算 $\mathbf{E}_i = \begin{pmatrix} 1 & 0 \\ r_i & s_i \end{pmatrix} \cdot \mathbf{D} \cdot \begin{pmatrix} 1 & t_i \\ 0 & 1 \end{pmatrix}$, 其中 $r_i, s_i, t_i \leftarrow \mathsf{U}(\mathbb{Z}_p)$.

如果 $\mathbf{D}$ 是一个 DDH 实例, 即 $\mathbf{D}_{1,1} = 1 \wedge \mathsf{rank}(\mathbf{D}) = 1$, 则 $\mathbf{E}_i$ 也是一个 DDH 实例.

假设 $\Rightarrow$ 假设 2: DDH 的随机自规约性

如何将一个 DDH 实例“扩展”成多个 DDH 实例?

DDH 的随机自规约

- 给定一个实例 $\mathbf{D} = \begin{bmatrix} 1 & b \\ a & c \end{bmatrix}$,
- 计算 $\mathbf{E}_i = \begin{pmatrix} 1 & 0 \\ r_i & s_i \end{pmatrix} \cdot \mathbf{D} \cdot \begin{pmatrix} 1 & t_i \\ 0 & 1 \end{pmatrix}$, 其中 $r_i, s_i, t_i \hookleftarrow \mathsf{U}(\mathbb{Z}_p)$.

如果 $\mathbf{D}$ 是一个 DDH 实例, 即 $\mathbf{D}_{1,1} = 1 \wedge \mathsf{rank}(\mathbf{D}) = 1$, 则 $\mathbf{E}_i$ 也是一个 DDH 实例.

定义 $\mathsf{RSR}_{r,s,t}(\mathbf{D}) = \begin{pmatrix} 1 & 0 \\ r & s \end{pmatrix} \cdot \mathbf{D} \cdot \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}$, 则当 $c \neq ab$ 时, $\mathsf{RSR}_{r,s,t}$ 为一个单射, 因此给定矩阵 $\mathbf{D}, \mathbf{E}$ 都为非 DDH 实例时, 存在唯一 r, s, t 满足 $\mathbf{E} = \mathsf{RSR}_{r,s,t}(\mathbf{D})$.

References

Diffie, W. and Hellman, M. E. (1976).
New directions in cryptography.
IEEE Transactions on Information Theory, 22(6):644–654.

ElGamal, T. (1985).
A public key cryptosystem and a signature scheme based on discrete logarithms.
IEEE Transactions on Information Theory, 31(4):469–472.

Rivest, R. L., Shamir, A., and Adleman, L. M. (1978).
A method for obtaining digital signatures and public-key cryptosystems.
Communications of the Association for Computing Machinery, 21(2):120–126.