基础知识

现代密码学与基础原语

《安全多方计算——可证明安全视角》第二章

2026 年 1 月 26 日

目录

现代密码学的三大原则

核心范式

相比于传统密码学的“艺术”与“直觉”，现代密码学基于严谨的科学范式：

1. **形式化的定义 (Formal Definitions)**：明确“安全保证”与“威胁模型”.
2. **精确的假设 (Precise Assumptions)**：如大整数分解、DDH 等数学难题.
3. **严格的安全性证明 (Rigorous Proofs)**：基于归约 (Reduction) 的证明.

1. 形式化的定义

安全保证 (Security Guarantee)

- 错误直觉：不能恢复明文.
- **正确定义**：敌手无法从密文中获得关于明文的**任何信息** (语义安全).

1. 形式化的定义

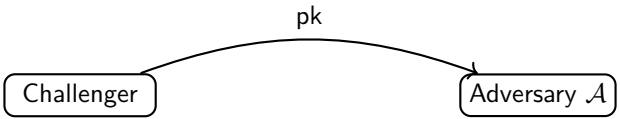
安全保证 (Security Guarantee)

- 错误直觉：不能恢复明文.
- **正确定义**：敌手无法从密文中获得关于明文的**任何信息** (语义安全).

威胁模型 (Threat Model)

- *Ciphertext-only*: 仅见密文.
- *Known-plaintext*: 已知部分明密文对.
- *Chosen-plaintext (CPA)*: 可选择明文加密.
- *Chosen-ciphertext (CCA)*: 可获得解密谕示.

基于游戏的安全性定义示例：IND-CPA



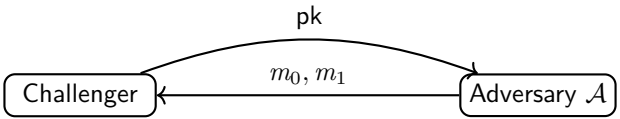
定义 (选择明文攻击下的不可区分游戏 INDis-tinguishability under Chosen-Plaintext Attack, IND-CPA)

若任意 PPT 敌手 $\mathcal{A}$ 赢得游戏的优势可忽略，即：

$$\Pr[b = b'] \leq \frac{1}{2} + \mathsf{negl}(\kappa)$$

则称该公钥加密方案是 IND-CPA 安全的.

基于游戏的安全性定义示例：IND-CPA



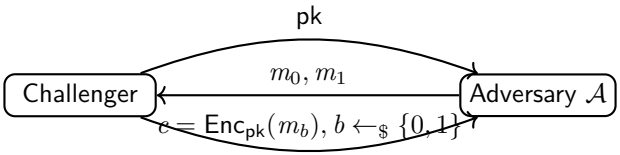
定义 (选择明文攻击下的不可区分游戏 INDis-tinguishability under Chosen-Plaintext Attack, IND-CPA)

若任意 PPT 敌手 $\mathcal{A}$ 赢得游戏的优势可忽略，即：

$$\Pr[b = b'] \leq \frac{1}{2} + \mathsf{negl}(\kappa)$$

则称该公钥加密方案是 IND-CPA 安全的.

基于游戏的安全性定义示例：IND-CPA



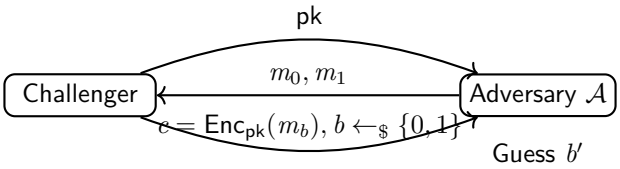
定义 (选择明文攻击下的不可区分游戏 INDis-tinguishability under Chosen-Plaintext Attack, IND-CPA)

若任意 PPT 敌手 $\mathcal{A}$ 赢得游戏的优势可忽略，即：

$$\Pr[b = b'] \leq \frac{1}{2} + \mathsf{negl}(\kappa)$$

则称该公钥加密方案是 IND-CPA 安全的.

基于游戏的安全性定义示例：IND-CPA



定义 (选择明文攻击下的不可区分游戏 INDis-tinguishability under Chosen-Plaintext Attack, IND-CPA)

若任意 PPT 敌手 $\mathcal{A}$ 赢得游戏的优势可忽略，即：

$$\Pr[b = b'] \leq \frac{1}{2} + \mathsf{negl}(\kappa)$$

则称该公钥加密方案是 IND-CPA 安全的.

2. 精确的假设

- 使用未经证明但久经考验的假设来证明密码方案的安全性.

2. 精确的假设

- 使用未经证明但久经考验的假设来证明密码方案的安全性.

DDH (Decisional Diffie-Hellman) 假设

如果对任意概率多项式时间的敌手 $\mathcal{A}$，都存在一个可忽略函数 negl，使得

$$|\Pr[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] - \Pr[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1]| \leq \mathsf{negl}(\kappa)$$

其中，群参数 $\mathbb{G}, q, g$ 是由群的生成算法 $\mathcal{G}(1^\kappa)$ 生成的，$x, y, z \leftarrow_\$ \mathbb{Z}_q$ 是均匀随机选择的. 那么，我们说 DDH 问题在群 $\mathbb{G}$ 中是困难的.

3. 严格的安全性证明

归约证明 (Reduction Proof) 的逻辑:

- **逆否命题**: 如果存在敌手 $\mathcal{A}$ 能攻破方案 Π(如 ElGamal 加密方案)，那么存在算法 $\mathcal{B}$ 能攻破底层假设 (如 DDH).

3. 严格的安全性证明

归约证明 (Reduction Proof) 的逻辑:

- **逆否命题**: 如果存在敌手 $\mathcal{A}$ 能攻破方案 Π(如 ElGamal 加密方案), 那么存在算法 $\mathcal{B}$ 能攻破底层假设 (如 DDH).

定义 (ElGamal 加密方案)

考虑一个阶为 q 的循环群 $\mathbb{G}$, 生成元为 g. ElGamal 加密方案包含三个算法 $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$:

- **密钥生成** Gen: 随机选择 $\mathsf{sk} \leftarrow_{\$} \mathbb{Z}_q$, 计算 $\mathsf{pk} = g^{\mathsf{sk}}$, 输出 $(\mathsf{pk}, \mathsf{sk})$.
- **加密** $\mathsf{Enc}(\mathsf{pk}, m)$: 对消息 $m \in \mathbb{G}$, 随机取 $r \leftarrow_{\$} \mathbb{Z}_q$, 输出密文 $c = (c_1, c_2) = (g^r, m \cdot \mathsf{pk}^r)$.
- **解密** $\mathsf{Dec}(\mathsf{sk}, c)$: 对 $c = (c_1, c_2)$, 输出 $\hat{m} = c_2 / c_1^{\mathsf{sk}}$ (/ 表示乘以群元素的逆元).

3. 严格的安全性证明

归约证明 (Reduction Proof) 的逻辑:

- **逆否命题**: 如果存在敌手 $\mathcal{A}$ 能攻破方案 Π(如 ElGamal 加密方案), 那么存在算法 $\mathcal{B}$ 能攻破底层假设 (如 DDH).

定义 (ElGamal 加密方案)

考虑一个阶为 q 的循环群 $\mathbb{G}$, 生成元为 g. ElGamal 加密方案包含三个算法 $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$:

- **密钥生成** Gen: 随机选择 $\mathsf{sk} \leftarrow_{\$} \mathbb{Z}_q$, 计算 $\mathsf{pk} = g^{\mathsf{sk}}$, 输出 $(\mathsf{pk}, \mathsf{sk})$.
- **加密** $\mathsf{Enc}(\mathsf{pk}, m)$: 对消息 $m \in \mathbb{G}$, 随机取 $r \leftarrow_{\$} \mathbb{Z}_q$, 输出密文 $c = (c_1, c_2) = (g^r, m \cdot \mathsf{pk}^r)$.
- **解密** $\mathsf{Dec}(\mathsf{sk}, c)$: 对 $c = (c_1, c_2)$, 输出 $\hat{m} = c_2 / c_1^{\mathsf{sk}}$ ($/$ 表示乘以群元素的逆元).

定理 (ElGamal 加密方案的安全性)

假设 *DDH* 问题在群 $\mathbb{G}$ 中是困难的, 那么 *ElGamal* 加密方案是 *IND-CPA* 安全的.

ElGamal 安全性证明（归约）

为严格证明 ElGamal 加密方案在 DDH 假设下的安全性，注意到以下逻辑关系：

(DDH 假设安全 $\Rightarrow$ElGamal 加密方案安全)$\Leftrightarrow$(ElGamal 加密方案不安全 $\Rightarrow$DDH 假设不安全)

ElGamal 安全性证明（归约）

为严格证明 ElGamal 加密方案在 DDH 假设下的安全性，注意到以下逻辑关系：

(DDH 假设安全 $\Rightarrow$ElGamal 加密方案安全)$\Leftrightarrow$(ElGamal 加密方案不安全 $\Rightarrow$DDH 假设不安全)

设 ElGamal 加密方案为 Π. 若存在敌手 $\mathcal{A}$ 攻破 IND-CPA：

$$\Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa)=1]=\frac{1}{2}+\epsilon(\kappa),$$

且 $\epsilon(\kappa)$ 非可忽略，则可构造算法 $\mathcal{B}$ 攻破 DDH.

ElGamal 安全性证明（归约）

为严格证明 ElGamal 加密方案在 DDH 假设下的安全性，注意到以下逻辑关系：

(DDH 假设安全 $\Rightarrow$ElGamal 加密方案安全)$\Leftrightarrow$(ElGamal 加密方案不安全 $\Rightarrow$DDH 假设不安全)

设 ElGamal 加密方案为 Π. 若存在敌手 $\mathcal{A}$ 攻破 IND-CPA：

$$\Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa)=1]=\frac{1}{2}+\epsilon(\kappa),$$

且 $\epsilon(\kappa)$ 非可忽略，则可构造算法 $\mathcal{B}$ 攻破 DDH.

算法 $\mathcal{B}$（输入 $(\mathbb{G},q,g,h_1,h_2,h_3)$）

- 设 $\mathsf{pk}=h_1$，发送给 $\mathcal{A}$，获得 $m_0,m_1\in\mathbb{G}$.
- 随机取 $b'\leftarrow_{\$}\{0,1\}$，设 $c_1=h_2$，$c_2=m_{b'}\cdot h_3$.
- 将 (c_1,c_2) 发给 $\mathcal{A}$ 得到 b；若 $b=b'$ 输出 1，否则输出 0.

ElGamal 安全性证明（归约）

情况 1：$(h_1, h_2, h_3) = (g^x, g^y, g^z)$ **为随机三元组**. **此时** (c_1, c_2) **与** m_b **无关**，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] = \frac{1}{2}.$$

ElGamal 安全性证明（归约）

情况 1：$(h_1, h_2, h_3) = (g^x, g^y, g^z)$ **为随机三元组**. **此时** (c_1, c_2) **与** m_b **无关**，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] = \frac{1}{2}.$$

情况 2：$(h_1, h_2, h_3) = (g^x, g^y, g^{xy})$ **为 DDH 三元组**. **此时分布与 IND-CPA 游戏一致**，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa) = 1].$$

ElGamal 安全性证明（归约）

情况 1：$(h_1, h_2, h_3) = (g^x, g^y, g^z)$ **为随机三元组**．**此时** (c_1, c_2) **与** m_b **无关**，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] = \frac{1}{2}.$$

情况 2：$(h_1, h_2, h_3) = (g^x, g^y, g^{xy})$ **为 DDH 三元组**．**此时分布与 IND-CPA 游戏一致**，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa) = 1].$$

综上所述，我们有

$$\begin{aligned}
&|\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] - \Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1]| \\
=&\left|\frac{1}{2} - \Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa) = 1]\right| = \epsilon(\kappa).
\end{aligned}$$

ElGamal 安全性证明（归约）

情况 1：$(h_1, h_2, h_3) = (g^x, g^y, g^z)$ 为随机三元组．此时 (c_1, c_2) 与 m_b 无关，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] = \frac{1}{2}.$$

情况 2：$(h_1, h_2, h_3) = (g^x, g^y, g^{xy})$ 为 DDH 三元组．此时分布与 IND-CPA 游戏一致，

$$\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa) = 1].$$

综上所述，我们有

$$\begin{aligned} &|\Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1] - \Pr[\mathcal{B}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1]| \\ =&\left|\frac{1}{2} - \Pr[\mathsf{PubK}^{\mathsf{CPA}}_{\mathcal{A},\Pi}(\kappa) = 1]\right| = \epsilon(\kappa). \end{aligned}$$

它不是可忽略函数，因此 $\mathcal{B}$ 破坏 DDH 假设，证毕．

基本术语与符号

- **可忽略函数 (Negligible Function)** $\mathsf{negl}(\kappa)$: 趋近于 0 的速度快于任何多项式的倒数.

$$\forall c > 0, \exists N, \forall \kappa > N, \mathsf{negl}(\kappa) < 1/\kappa^c$$

基本术语与符号

- **可忽略函数 (Negligible Function)** $\mathsf{negl}(\kappa)$: 趋近于 0 的速度快于任何多项式的倒数.

$$\forall c > 0, \exists N, \forall \kappa > N, \mathsf{negl}(\kappa) < 1/\kappa^c$$

- **计算不可区分 (Computational Indistinguishability)** $\approx$: 对于任意 PPT 区分器 D，区分两个分布 X, Y 的概率差为可忽略函数.

$$|\Pr[D(X) = 1] - \Pr[D(Y) = 1]| \leq \mathsf{negl}(\kappa)$$

基本术语与符号

- **可忽略函数 (Negligible Function)** $\mathsf{negl}(\kappa)$: 趋近于 0 的速度快于任何多项式的倒数.

$$\forall c > 0, \exists N, \forall \kappa > N, \mathsf{negl}(\kappa) < 1/\kappa^c$$

- **计算不可区分 (Computational Indistinguishability)** $\approx$: 对于任意 PPT 区分器 D，区分两个分布 X, Y 的概率差为可忽略函数.

$$|\Pr[D(X) = 1] - \Pr[D(Y) = 1]| \leq \mathsf{negl}(\kappa)$$

- **统计距离 (Statistical Distance)**:

$$\delta(X, Y) = \frac{1}{2} \sum_d |\Pr[X = d] - \Pr[Y = d]|$$

1. 门限秘密分享 (Threshold Secret Sharing)

定义：设 $\mathcal{D}$ 是秘密所在的域，$\mathcal{D}_1$ 是份额所在的域. (t,n)-门限秘密分享包含算法 $(\mathsf{Share},\mathsf{Rec})$：

- $(s_1,\ldots,s_n)\leftarrow\mathsf{Share}(s)$：输入 $s\in\mathcal{D}$，输出 n 个份额 $s_1,\ldots,s_n\in\mathcal{D}_1$.
- $s=\mathsf{Rec}(s_{i_1},\ldots,s_{i_k})$：输入任意 $k\geq t+1$ 个份额，输出 $s\in\mathcal{D}$.

1. 门限秘密分享 (Threshold Secret Sharing)

定义：设 $\mathcal{D}$ 是秘密所在的域，$\mathcal{D}_1$ 是份额所在的域. (t,n)-门限秘密分享包含算法 $(\mathsf{Share},\mathsf{Rec})$：

- $(s_1,\ldots,s_n)\leftarrow\mathsf{Share}(s)$：输入 $s\in\mathcal{D}$，输出 n 个份额 $s_1,\ldots,s_n\in\mathcal{D}_1$.
- $s=\mathsf{Rec}(s_{i_1},\ldots,s_{i_k})$：输入任意 $k\geq t+1$ 个份额，输出 $s\in\mathcal{D}$.

性质：

- **正确性**：若 $(s_1,\ldots,s_n)\leftarrow\mathsf{Share}(s)$，则

$$\Pr[\forall k\geq t+1,\mathsf{Rec}(s_{i_1},\ldots,s_{i_k})=s]=1.$$

- **（完美）隐私性**：任意 $|U|\leq t$ 的份额集合 $\{s_j\}_{j\in U}$ 的分布与 s 无关.

1. 门限秘密分享 (Threshold Secret Sharing)

定义：设 $\mathcal{D}$ 是秘密所在的域，$\mathcal{D}_1$ 是份额所在的域. (t,n)-门限秘密分享包含算法 $(\mathsf{Share},\mathsf{Rec})$：

- $(s_1,\ldots,s_n)\leftarrow\mathsf{Share}(s)$：输入 $s\in\mathcal{D}$，输出 n 个份额 $s_1,\ldots,s_n\in\mathcal{D}_1$.
- $s=\mathsf{Rec}(s_{i_1},\ldots,s_{i_k})$：输入任意 $k\geq t+1$ 个份额，输出 $s\in\mathcal{D}$.

性质：

- **正确性**：若 $(s_1,\ldots,s_n)\leftarrow\mathsf{Share}(s)$，则

$$\Pr[\forall k\geq t+1,\mathsf{Rec}(s_{i_1},\ldots,s_{i_k})=s]=1.$$

- **(完美) 隐私性**：任意 $|U|\leq t$ 的份额集合 $\{s_j\}_{j\in U}$ 的分布与 s 无关.

Shamir Secret Sharing (基于多项式插值)

构造 t 次多项式 $f(x)=s+a_1x+\cdots+a_tx^t$. 份额为 $y_i=f(i)$.

2. 哈希函数与随机谕示机 (Hash Functions& Random Oracle)

哈希函数 $H\colon \{0,1\}^* \to \{0,1\}^\kappa$:

- **抗原像性**: 给定 y，找到 x' 使 $H(x') = y$ 在计算上不可行.
- **抗第二原像性**: 给定 x，找 $x' \neq x$ 使 $H(x) = H(x')$ 在计算上不可行.
- **抗碰撞性**: 找到 $x \neq x'$ 使 $H(x) = H(x')$ 在计算上不可行.

2. 哈希函数与随机谕示机 (Hash Functions& Random Oracle)

哈希函数 $H\colon \{0,1\}^* \to \{0,1\}^\kappa$:

- 抗原像性: 给定 y，找到 x' 使 $H(x') = y$ 在计算上不可行.
- 抗第二原像性: 给定 x，找 $x' \neq x$ 使 $H(x) = H(x')$ 在计算上不可行.
- 抗碰撞性: 找到 $x \neq x'$ 使 $H(x) = H(x')$ 在计算上不可行.

随机谕示机 (RO):

- 理想化模型.
- 公共函数 H 维护查询表；新输入返回随机 $r_x \leftarrow_\$ \{0,1\}^\kappa$，旧输入返回同一 r_x.
- 实际中用哈希函数实例化（启发式），可能与 RO 有差距.

3. 伪随机数生成器 (PseudoRandom Generator, PRG)

定义: 确定性算法 $G : \{0,1\}^{\kappa} \to \{0,1\}^{\ell(\kappa)}$, 其中 $\ell(\kappa) > \kappa$.

- **扩展性**: 输出长度大于输入种子.
- **伪随机性**: 输出与真随机串**计算不可区分**.

$$\{G(s)\}_{s \leftarrow \{0,1\}^{\kappa}} \approx \{r\}_{r \leftarrow \{0,1\}^{\ell(\kappa)}}$$

4. 对称加密方案 (Symmetric Encryption Schemes)

对称加密 (Symmetric Encryption)

设密钥空间为 $\mathcal{K}$，明文空间为 $\mathcal{M}$，密文空间为 $\mathcal{E}$. 对称加密方案包含算法 $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $c \leftarrow \mathsf{Enc}_\mathsf{k}(m)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $m \in \mathcal{M}$，输出 $c \in \mathcal{E}$.
- $m \leftarrow \mathsf{Dec}_\mathsf{k}(c)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $c \in \mathcal{E}$，输出 $m \in \mathcal{M} \cup \{\perp\}$.

4. 对称加密方案 (Symmetric Encryption Schemes)

对称加密 (Symmetric Encryption)

设密钥空间为 $\mathcal{K}$，明文空间为 $\mathcal{M}$，密文空间为 $\mathcal{E}$. 对称加密方案包含算法 $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $c \leftarrow \mathsf{Enc}_\mathsf{k}(m)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $m \in \mathcal{M}$，输出 $c \in \mathcal{E}$.
- $m \leftarrow \mathsf{Dec}_\mathsf{k}(c)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $c \in \mathcal{E}$，输出 $m \in \mathcal{M} \cup \{\perp\}$.

正确性：对任意 $k \leftarrow \mathsf{Gen}(1^\kappa)$ 和 $m \in \mathcal{M}$，有 $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$.

4. 对称加密方案 (Symmetric Encryption Schemes)

对称加密 (Symmetric Encryption)

设密钥空间为 $\mathcal{K}$，明文空间为 $\mathcal{M}$，密文空间为 $\mathcal{E}$. 对称加密方案包含算法 $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $c \leftarrow \mathsf{Enc}_\mathsf{k}(m)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $m \in \mathcal{M}$，输出 $c \in \mathcal{E}$.
- $m \leftarrow \mathsf{Dec}_\mathsf{k}(c)$：输入 $\mathsf{k} \in \mathcal{K}$ 与 $c \in \mathcal{E}$，输出 $m \in \mathcal{M} \cup \{\perp\}$.

正确性：对任意 $k \leftarrow \mathsf{Gen}(1^\kappa)$ 和 $m \in \mathcal{M}$，有 $\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$.
IND-CPA 安全性：若对任意 PPT 敌手 $\mathcal{A}$，

$$\Pr\left[\begin{array}{l} k \leftarrow \mathsf{Gen}(1^\kappa); (m_0, m_1) \leftarrow \mathcal{A}^{\mathsf{Enc}_k(\cdot)}(1^\kappa); \\ b \leftarrow \{0,1\}; c \leftarrow \mathsf{Enc}_k(m_b); \\ b' \leftarrow \mathcal{A}^{\mathsf{Enc}_k(\cdot)}(1^\kappa, c, m_0, m_1) \end{array} : b = b'\right] \leq \frac{1}{2} + \mathsf{negl}(\kappa),$$

则称 $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ 为 IND-CPA 安全. 其中 $\mathsf{Enc}_k(\cdot)$ 为 $\mathcal{A}$ 可自适应查询的加密接口.
实际中需使用随机 IV 或概率加密以达到 IND-CPA.

5. 消息认证码 (Message Authentication Codes, MACs)

消息认证码 (MAC)

设密钥空间为 $\mathcal{K}$，消息空间为 $\mathcal{M}$，标签空间为 $\mathcal{T}$. MAC 方案包含算法 $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Ver})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $t \leftarrow \mathsf{Mac}_\mathsf{k}(m)$：输入 k 与 $m \in \mathcal{M}$，输出标签 $t \in \mathcal{T}$.
- $0/1 \leftarrow \mathsf{Ver}_\mathsf{k}(m, t)$：验证标签合法性.

正确性：对任意 $k \leftarrow \mathsf{Gen}(1^\kappa)$ 与 $m \in \mathcal{M}$，有 $\mathsf{Ver}_k(m, \mathsf{Mac}_k(m)) = 1$.

5. 消息认证码 (Message Authentication Codes, MACs)

消息认证码 (MAC)

设密钥空间为 $\mathcal{K}$，消息空间为 $\mathcal{M}$，标签空间为 $\mathcal{T}$. MAC 方案包含算法 $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Ver})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $t \leftarrow \mathsf{Mac}_\mathsf{k}(m)$：输入 k 与 $m \in \mathcal{M}$，输出标签 $t \in \mathcal{T}$.
- $0/1 \leftarrow \mathsf{Ver}_\mathsf{k}(m, t)$：验证标签合法性.

正确性：对任意 $k \leftarrow \mathsf{Gen}(1^\kappa)$ 与 $m \in \mathcal{M}$，有 $\mathsf{Ver}_k(m, \mathsf{Mac}_k(m)) = 1$.

若对任意敌手 $\mathcal{A}$，

$$\Pr\left[\begin{array}{l}\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa);\ m' \leftarrow \mathcal{A}(1^\kappa);\\ t' \leftarrow \mathsf{Mac}_\mathsf{k}(m');\ (m,t) \leftarrow \mathcal{A}(1^\kappa, m', t')\end{array} : \mathsf{Ver}_\mathsf{k}(m,t) = 1 \ \wedge\ m \neq m'\right] \leq \mathsf{negl}(\kappa),$$

则称 $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Ver})$ 为信息论安全的一次性 MAC.

5. 消息认证码 (Message Authentication Codes, MACs)

消息认证码 (MAC)

设密钥空间为 $\mathcal{K}$，消息空间为 $\mathcal{M}$，标签空间为 $\mathcal{T}$. MAC 方案包含算法 $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Ver})$：

- $\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa)$：输出密钥 $\mathsf{k} \in \mathcal{K}$.
- $t \leftarrow \mathsf{Mac}_\mathsf{k}(m)$：输入 k 与 $m \in \mathcal{M}$，输出标签 $t \in \mathcal{T}$.
- $0/1 \leftarrow \mathsf{Ver}_\mathsf{k}(m, t)$：验证标签合法性.

正确性：对任意 $k \leftarrow \mathsf{Gen}(1^\kappa)$ 与 $m \in \mathcal{M}$，有 $\mathsf{Ver}_k(m, \mathsf{Mac}_k(m)) = 1$.

若对任意敌手 $\mathcal{A}$，

$$\Pr\left[\begin{array}{l}\mathsf{k} \leftarrow \mathsf{Gen}(1^\kappa);\ m' \leftarrow \mathcal{A}(1^\kappa);\\ t' \leftarrow \mathsf{Mac}_\mathsf{k}(m');\ (m,t) \leftarrow \mathcal{A}(1^\kappa, m', t')\end{array} : \mathsf{Ver}_\mathsf{k}(m,t) = 1 \ \wedge \ m \neq m'\right] \leq \mathsf{negl}(\kappa),$$

则称 $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Ver})$ 为信息论安全的一次性 MAC.

线性一次性 MAC: 令 $\mathcal{M} = \mathbb{Z}_p$，$\mathcal{T} = \mathbb{Z}_p$，密钥 $(a, b) \leftarrow \mathbb{Z}_p^2$. $\mathsf{Mac}_{a,b}(m) = a \cdot m + b \pmod p$.

6. 承诺 (Commitments)

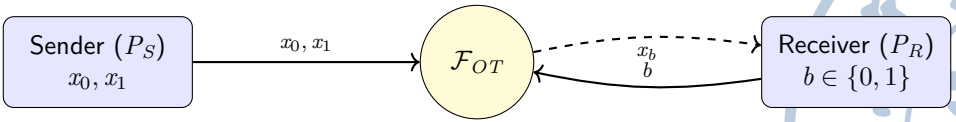
“把消息放入信封，稍后打开.”

1. **隐藏性 (Hiding)**: 接收方在打开前无法得知 m.
2. **绑定性 (Binding)**: 承诺方无法将 c 打开为 $m' \neq m$.

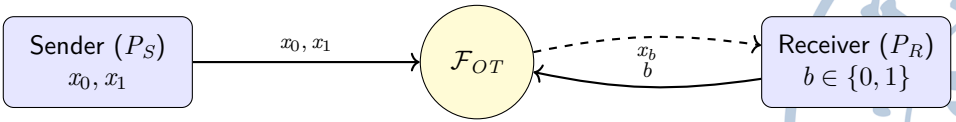
常见构造:

- **Hash-based**: $c = H(m||r)$ (需 RO 模型).
- **Pedersen**: $c = g^m h^r$ (基于离散对数，完美隐藏，计算绑定).

7. 茫然传输 (Oblivious Transfer, OT)



7. 茫然传输 (Oblivious Transfer, OT)



性质:

- **接收方隐私**: 发送方不知道 b.
- **发送方隐私**: 接收方不知道 x_{1-b}.

本章思考题

1. 可证明安全的协议在现实中一定安全吗？(考虑侧信道、实现漏洞)
2. 密码学游戏与日常游戏的共同点？(规则、交互、胜负条件)
3. 随机谕示机 vs 现实哈希函数？
4. 若 $P = NP$，是否存在 PRG？(单向函数是基础)

Q & A