# 密码学复杂性的一点注记

---

**吉勒·布拉萨德（Gilles Brassard）**

**摘要**：本文为以下观点提供了证据：基于单向函数（例如 Diffie 与 Hellman 所提出的那种）的密码系统的计算安全性，其证明本身可能是困难的。若能证明密码分析任务是 NP 完全的，则将意味着 $\mathbf{NP} = \mathbf{CoNP}$。

---

## 一、引言

Diffie 与 Hellman [1] 提议在有限域中使用指数函数进行密码学应用。该提议基于如下猜想：其反函数——对数函数——在计算上不可行。对该猜想最好的间接支持，就是证明对数问题是 **NP-难** 的，即存在一台以对数函数为预言机的图灵机，能在多项式时间内判定某个 NP 完全集合。如果确实如此，那么在普遍接受的假设 $P \neq \mathbf{NP}$ [2, 第10章] 下，我们就能确信对数问题确实是困难的。

但这里需要谨慎。传统的复杂性理论关注的是**最坏情况**下的行为：若对任意计算某函数 $f$ 的算法 $A$ 和任意多项式 $P$，总存在某个输入 $x$（长度为 $n$），使得 $A$ 在 $x$ 上的运行时间超过 $P(n)$，则称 $f$ 不可高效计算。然而，密码学所需的概念与此不同：一个允许敌方轻易破译绝大多数密文的密码系统显然是糟糕的。尽管如此，即使要证明对数问题在最坏情况下是困难的，目前的技术也尚无法做到，更不用说证明 Diffie-Hellman 密码系统的计算安全性了。

我们称一台非确定性图灵机 $M$ **计算函数** $f$，是指对任意输入 $x$，至少存在一条计算路径停机，且所有停机路径的输出带内容均为 $f(x)$。类似地，$M$ **接受集合** $S$，是指对任意输入 $x$，存在一条停机（即接受）的计算路径当且仅当 $x \in S$。关键观察在于：使用这样的机器，可以在多项式时间内计算有限域上的对数。作为 Miller 定理 [3] 的一个应用，对数函数的**投影** $P_{\log}$（定义见下文）属于 $\mathbf{NP} \cap \mathbf{CoNP}$。此外，$P_{\log} \in P$ 当且仅当对数可在确定性多项式时间内计算。因此，若 Diffie 与 Hellman 的猜想成立，则意味着 $P \subsetneq \mathbf{NP} \cap \mathbf{CoNP}$，而 $P_{\log}$ 即为其见证。进一步地，若对数问题是 NP-难的，我们将得出 $\mathbf{NP} = \mathbf{CoNP}$，因为此时 $P_{\log}$ 将既是 NP-完全的，又属于 CoNP。

由于上述两个结论（即 $P \subsetneq \mathbf{NP} \cap \mathbf{CoNP}$ 与 $\mathbf{NP} = \mathbf{CoNP}$）要么被认为是错误的，要么极难证明，我们由此推断：Diffie-Hellman 密码系统的安全性同样难以被严格证明。

---

## 二、构造

为使本节自包含，我们将不显式引用 Miller 的结果 [3]，但所用技术与其类似。我们还将简化“对数可在非确定性图灵机上多项式时间计算”这一事实的证明。

定义对数函数 $\log(p, r, n)$ 如下：若 $p$ 为素数，$r$ 是 $p$ 的一个原根 [4, 第4.5.4节]，且 $0 < n < p$，则 $\log(p, r, n)$ 是满足 $0 < m < p$ 且 $r^m \equiv n \pmod{p}$ 的唯一整数 $m$；否则 $\log(p, r, n) = 0$。

定义对数函数的**投影** $P_{\log}$ 为：Missing or unrecognized delimiter for \left 其中 $\langle p, r, n, t \rangle$ 是 $\langle p, \langle r, \langle n, t \rangle \rangle \rangle$ 的简写，$\langle \cdot, \cdot \rangle$ 为某种合适的配对函数。

- $P_{\log} \in \mathbf{NP}$：
$p$ 是素数且 $r$ 是 $p$ 的原根，当且仅当 $r^{p-1} \equiv 1 \pmod{p}$，且对 $p-1$ 的每个素因子 $q$，均有 $r^{(p-1)/q} \not\equiv 1 \pmod{p}$ [4, 第4.5.4节]。这些条件可通过非确定性多项式时间算法验证：先猜测 $p-1$ 的素因子分解，并为每个因子提供素性证书 [5]，再使用分治策略计算模 $p$ 的幂 [4, 第4.6.3节]。一旦确认 $p$ 为素数且 $r$ 为原根（并假设 $0 < n < p$），即可猜测唯一的 $m$（满足 $0 < m < p$ 且 $r^m \equiv n \pmod{p}$）。若 $m > t$，则 $\langle p, r, n, t \rangle \in P_{\log}$。
- $P_{\log} \in \mathbf{CoNP}$：
$\langle p, r, n, t \rangle \notin P_{\log}$ 当且仅当 $\log(p, r, n) \leqslant t$。这等价于以下任一情形成立：存在 $i, j$（$0 < i < j < p$）使得 $r^i \equiv r^j \pmod{p}$；存在 $m \leqslant t$ 使得 $r^m \equiv n \pmod{p}$；或 $n$ 不满足 $0 < n < p$。这些条件均可在非确定性多项式时间内验证。
- **若 $P_{\log} \in P$，则对数可在确定性多项式时间内计算：**
可使用如下二分搜索算法：

```
function log(p, r, n)
    i := 0
    j := p - 1
    while i < j do
        /* 此时有 i ≤ log ≤ j */
        k := floor((i + j) / 2)
        if ⟨p, r, n, k⟩ ∈ P_log then
            i := k + 1
        else
            j := k
        fi
    od
    return i
end log
```

- **若对数问题是 NP-难的，则 $\mathbf{NP} = \mathbf{CoNP}$：**
由上述算法可知，给定 $P_{\log}$ 的预言机即可高效计算对数，故对数问题的 NP-难度蕴含 $P_{\log}$ 的 NP-难度。由于 $P_{\log} \in \mathbf{NP}$，它将是 NP-完全的。但同时 $P_{\log} \in \mathbf{CoNP}$。我们由此得出 $\mathbf{NP} = \mathbf{CoNP}$，因为否则不可能存在一个 NP-完全集合，其补集也在 NP 中（详见附录证明）。

---

## 三、推广

上述推理可推广至其他基于单向函数的密码系统。如 [1] 中所定义，**单向函数** $f$ 满足：对定义域内任意输入 $x$，计算 $f(x)$ 是容易的；但对值域中几乎所有 $y$，求解方程 $y = f(x)$ 在计算上是不可行的。迄今为止，我们使用的是 Diffie 与 Hellman 提出的候选单向函数——有限域上的指数运算。

事实上，若任一满足若干限制条件的函数 $f$ 被证明是单向的，则将推出 $P \nsubseteq \mathbf{NP} \cap \mathbf{CoNP}$。类似地，若通过证明 $f$ 的逆问题是 NP-难的来佐证其单向性，则将推出 $\mathbf{NP} = \mathbf{CoNP}$。这些限制条件包括：$f$ 是从一个 NP 集合到一个 CoNP 集合的单射，且存在多项式 $P$，使得对 $f$ 值域中的任意 $x$，均有 $|x| < P(|f(x)|)$（其中 $|x|$ 表示 $x$ 的二进制表示长度）。

最后举一例：Rivest、Shamir 与 Adleman [6] 提出了一种公钥密码系统，其安全性基于如下信念：**在素数集上，乘法是单向的**。更通俗地说，将两个素数相乘很容易，但我们尚不知道如何高效地分解一个大整数（即使已知它是两个素数的乘积）。同样地，若能证明该密码系统的计算安全性，则意味着 $P \nsubseteq \mathbf{NP} \cap \mathbf{CoNP}$；若能证明其密码分析问题是 NP-难

的，则意味着 $\mathbf{NP} = \mathbf{CoNP}$。

为直接证明这些结论，可模仿第二节的证明，将 $P_{\log}$ 替换为集合 $S$：

Missing or unrecognized delimiter for \left 如同第二节，可证明 $S \in \mathbf{NP} \cap \mathbf{CoNP}$，且 $S \in P$ 当且仅当素因子分解可在多项式时间内完成。

---


## 致谢

Leonard Adleman、Ronald Rivest 与 Gary Miller [7] 独立得到了类似结果。


---

## 附录

我们将证明：若集合 $A$ 是 NP-完全的，且 $A \in \mathbf{CoNP}$，则 $\mathbf{NP} = \mathbf{CoNP}$。对于 Karp 的 NP-完全性定义 [8] 这是显然的，但对于本文所采用的 Cook 定义 [9] 则需证明。

设 $M_+$ 与 $M_-$ 分别为接受 $A$ 及其补集的非确定性多项式时间图灵机。对任意 $L \in \mathbf{NP}$，存在一台确定性多项式时间图灵机 $M$，它借助 $A$ 的预言机可接受 $L$。考虑非确定性图灵机 $M'$，它在输入上模拟 $M$ 的每一步，但当 $M$ 向预言机提问时，$M'$ 非确定性地猜测答案应为“是”或“否”，并相应地模拟 $M_+$ 或 $M_-$。若 $M'$ 到达 $M_+$ 或 $M_-$ 的接受状态，则继续按正确答案模拟 $M$；若到达拒绝状态，则立即拒绝原输入。当 $M$ 到达终止状态时，$M'$ 接受当且仅当 $M$ 拒绝。

显然，$M'$ 在多项式时间内运行，且接受 $L$ 的补集，故 $L \in \mathbf{CoNP}$。这证明了 $\mathbf{NP} \subseteq \mathbf{CoNP}$。另一方面，对任意 $L \in \mathbf{CoNP}$，其补集属于 $\mathbf{NP}$，从而也属于 $\mathbf{CoNP}$（因 $\mathbf{NP} \subseteq \mathbf{CoNP}$），故 $L \in \mathbf{NP}$。因此 $\mathbf{CoNP} \subseteq \mathbf{NP}$，得证 $\mathbf{NP} = \mathbf{CoNP}$。

---

## 参考文献


[1] W. Diffie and M. E. Hellman, “New directions in cryptography,” *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 644–654, Nov. 1976.
[2] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974.
[3] G. L. Miller, “Riemann's hypothesis and tests for primality,” *J. Comput. Syst. Sci.*, vol. 13, pp. 300–317, 1976.
[4] D. E. Knuth, *The Art of Computer Programming*, vol. 2, *Semi-Numerical Algorithms*. Reading, MA: Addison-Wesley, 1969.
[5] V. Pratt, "Every prime has a succinct certificate," *SIAM J. Comput.*, vol. 4, pp. 214–220, 1975.
[6] R. Rivest, A. Shamir, and L. Adleman, “A method of obtaining digital signatures and public-key cryptosystem,” *Commun. ACM*, vol. 21, pp. 120–126, Feb. 1978.
[7] L. Adleman, private communication, 1978.
[8] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. E. Miller and J. W. Thatcher, Eds. NY: Plenum, 1972.
[9] S. A. Cook, “The complexity of theorem proving procedures,” in *Proc. 3rd Ann. ACM Symp. Theory of Computing*, 1971, pp. 151–158.