第 2 章：分组密码体制

2.6 多重加密

赵俊舟

junzhou.zhao@xjtu.edu.cn

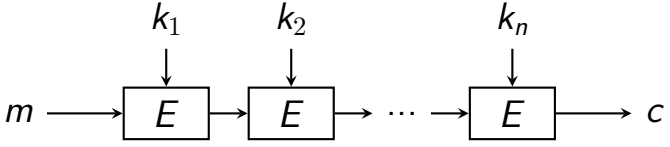
2025 年 3 月 12 日

目录

目录

多重加密提出的背景

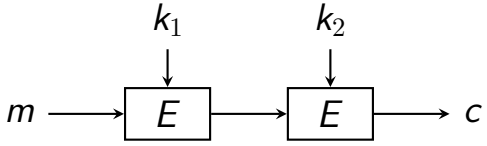
- DES 的密钥长度为 56 位，已经不安全，需要寻找更安全的加密方法。
- 例如，DES 密钥的穷举攻击目前仅需要 10 小时。
- 在高级加密标准 AES 出现之前，多重加密是一种增强加密算法安全性的解决方案。
- 即多次使用同一个加密算法，对明文反复加密。
- 多重加密的优点：可以利用现有软硬件资源。



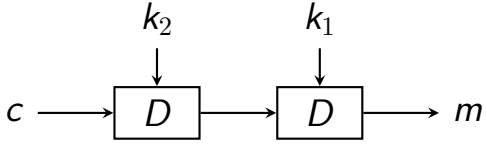
目录

双重加密与 2DES

- 多重加密最简单的形式是双重加密，使用两个密钥加密。
- 加密：$c = E(k_2, E(k_1, m))$

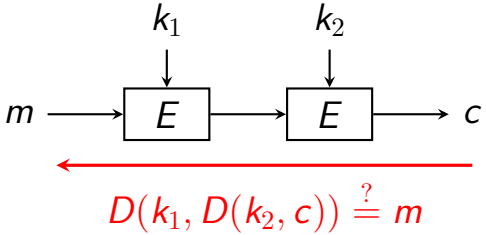


- 解密：$m = D(k_1, D(k_2, c))$



- 对于 DES，使用双重加密的 2DES 密钥长度为 112 位。

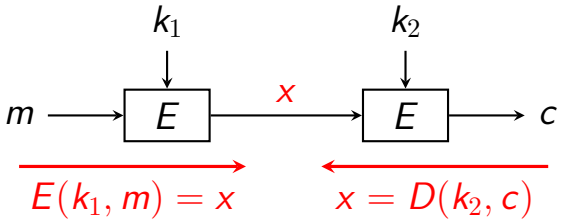
对双重加密的穷举攻击



- 给定密文 c，依次尝试所有可能的密钥 k_2 和 k_1，直到发现明文 m。
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次。
- 有没有更有效的攻击手段？

中间相遇攻击（Meet-in-the-Middle Attack）

- 中间相遇攻击对任何使用双重加密的分组密码都有效。

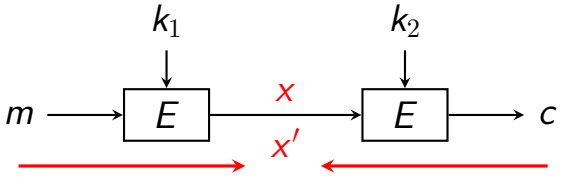


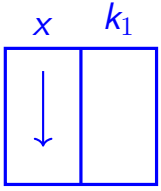

```
for k1 in K
    for k2 in K
        // ...
```

$\Rightarrow$

```
for k1 in K
    // ...
for k2 in K
    // ...
```

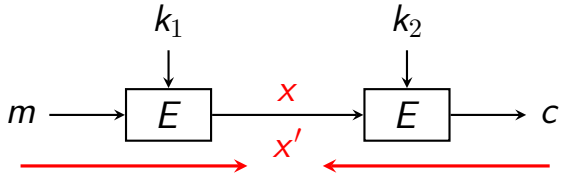
中间相遇攻击的攻击步骤

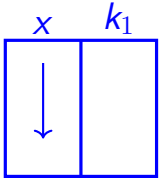




1. 搜集尽可能多的明密文对 (m, c)；
2. 遍历密钥 k_1 对 m 加密，得到的结果按 x 排序后保存在表中；
3. 遍历密钥 k_2 对 c 解密，每解一次密，在表中匹配；
4. 如产生匹配，则找到一对可能密钥，然后用这两个密钥对一个新的明密文对进行验证，若通过则说明密钥正确。

中间相遇攻击攻击 2DES 的复杂度分析





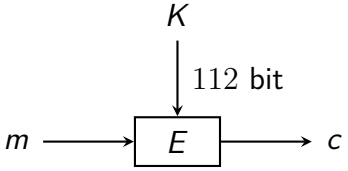
- 第 2 步和第 3 步的时间复杂度：$2^{56} + 2^{56} = 2^{57}$
- 空间复杂度：$(56 + 64) \times 2^{56}$ bit
- 第 4 步分析：
  - 使用一对 (m, c) 找到的错误密钥平均个数为：$2^{112}/2^{64} = 2^{48}$
  - 使用两对 (m, c) 找到的错误密钥平均个数为：$2^{48}/2^{64} = 2^{-16} \approx 0$

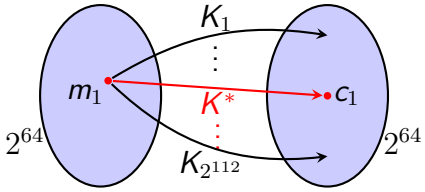
2DES 的安全性

相较于攻击 DES 的最差时间复杂度 2^{56}，2DES 的加密强度并没有提高很多！

中间相遇攻击第 4 步需要试多少对 (m, c)？

- 给定一个明密文对 (m_1, c_1)，分组长度 64 位，2DES 密钥长度 112 位，所以会存在 2^{112} 种映射将 m_1 映射为密文。
- 由于密文空间大小仅仅为 2^{64}，所以平均会有 $2^{112}/2^{64} = 2^{48}$ 种映射将 m_1 映射为 c_1。
- 即每个明密文对平均存在 2^{48} 个映射，其中 $2^{48} - 1 \approx 2^{48}$ 个映射是错误的。
- 使用第二个明密文对 (m_2, c_2) 验证时，会从 2^{48} 个映射中找到正确的映射，找到错误映射的平均个数为 $2^{48}/2^{64} = 2^{-16} \approx 0$。

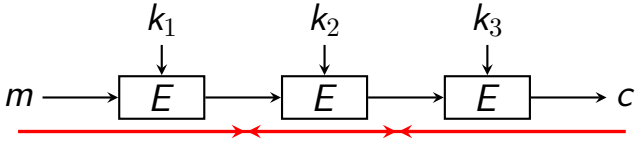




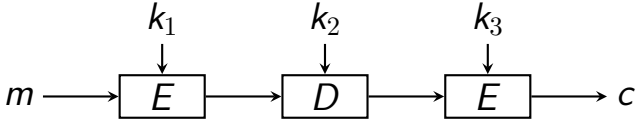
目录

三重加密与 3DES

- 为了进一步抵抗密码分析，可以使用三重加密：



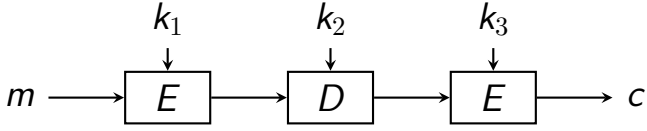
- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$。
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：



  - $k_1 \neq k_2 \neq k_3$ 强三重加密，密钥长度 $3 \times 56 = 168$
  - $k_1 = k_3 \neq k_2$ 普通三重加密，密钥长度 $2 \times 56 = 112$
  - $k_1 = k_2 = k_3$ 等价于普通分组加密，密钥长度 56

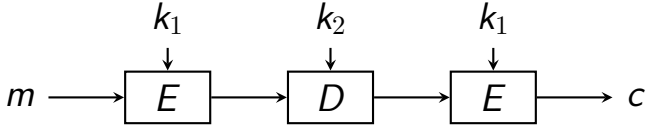
3DES 现状

- 使用工作模式 1 的 3DES 目前仍然被广泛应用；



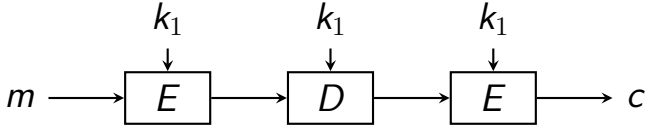
安全

- 使用工作模式 2 的 3DES 于 2017 年被认为已过时；



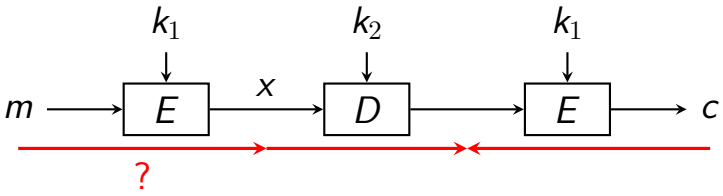
已过时

- 使用工作模式 3 的 3DES 等价于普通 DES，不安全。



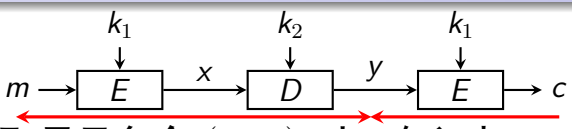
不安全

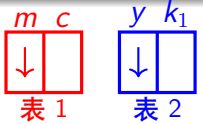
针对 3DES 的已知明文攻击



- 如果已知 x 和 c，那么对三重加密的攻击可以转化为对二重加密的攻击；
- 当然，只要攻击者不知道密钥 k_1，即使知道 m 和 c，还是无法知道 x；
- 然而，攻击者可以选择 x 的一个可能值，再试着找到一个可产生 x 的 (m, c) 对，从而将对三重加密的攻击转化为对二重加密的攻击。

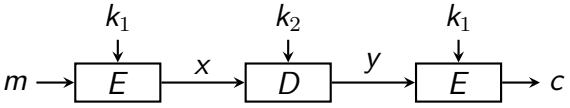
攻击步骤

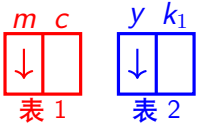




1. 获取尽量多个 (m, c) 对，存入表 1；
2. 随意选择值 x，按以下步骤创建表 2：
   - 对每个可能密钥 k_1 计算可产生 x 的明文 $m = D(k_1, x)$；
   - 在表 1 中匹配 m，若匹配成功，则在表 2 中添加一项 (y, k_1)，其中 $y = D(k_1, c)$。
3. 搜索 k_2：
   - 对每个可能密钥 k_2，计算 $y = D(k_2, x)$；
   - 在表 2 中匹配 y，若匹配成功，则找到一对密钥 (k_1, k_2) 可以产生已知 (m, c) 对。
4. 在其他 (m, c) 上验证找到的密钥对，若验证成功，则找到正确密钥对 (k_1, k_2)；否则，返回步骤 2。

时间复杂度分析





- 对给定的 (m, c) 对，选择 x 成功的可能性为 2^{-64}。
- 给定 n 个 (m, c) 对，则选择 x 成功的可能性为 $n2^{-64}$。
- 所以，前两步平均需要尝试 $2^{64}/n$ 次才会得到一个正确的 x。
- 对于每个 x，第 3 步还需要遍历 k_2，因此总的时间复杂度为 $2^{56}2^{64}/n = 2^{120-\log_2 n}$。

小结

1. 多重加密提出的背景
2. 双重加密及其安全性分析
3. 三重加密及其安全性分析