多重加密及其安全性分析

多重加密提出的背景

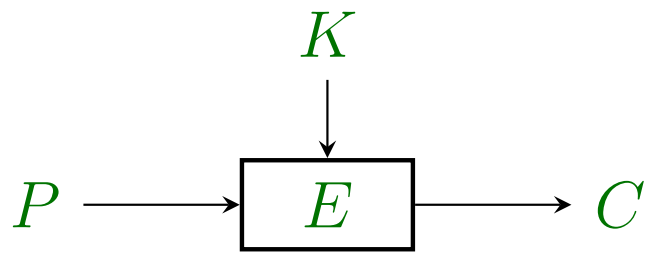
- DES 的密钥长度为 56 位，已经不安全，需要寻找更安全的加密方法。
  - 例如，DES 密钥的穷举攻击目前仅需要 10 小时！

- 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。
  - 将一个加密算法多次使用，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。

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！

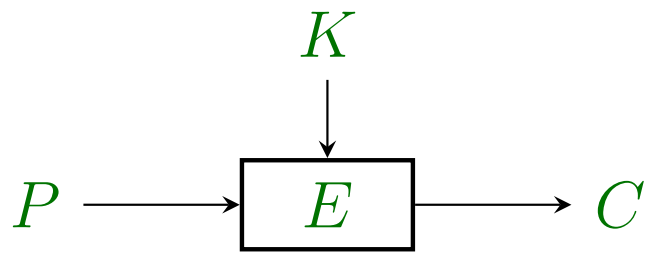


- 在高级加密标准 AES 出现之前，多重加密是一种增强加密算法安全性的解决方案。
  - 将一个加密算法多次使用，对明文反复加密。

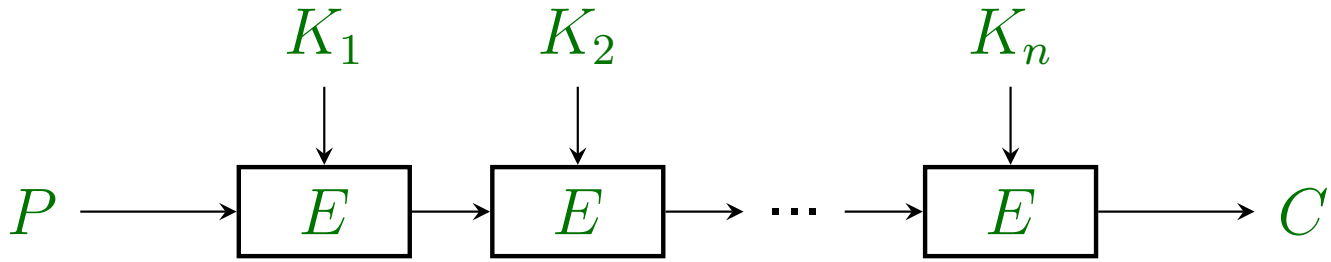
- 多重加密的优点：可以利用现有软硬件资源。

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！



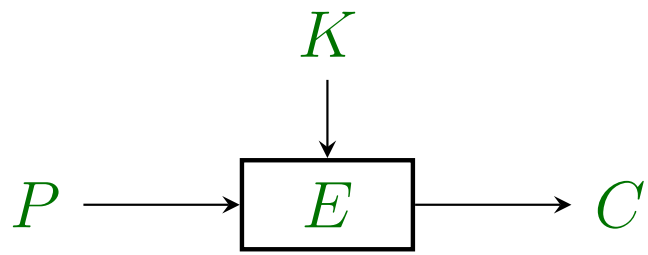
- 在高级加密标准 AES 出现之前，多重加密是一种增强加密算法安全性的解决方案。
  - 将一个加密算法多次使用，对明文反复加密。



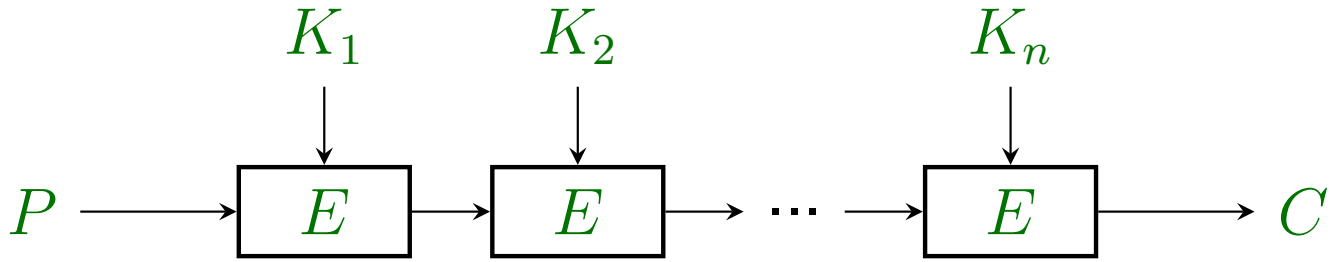
- 多重加密的优点：可以利用现有软硬件资源。

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。
  - 将一个加密算法多次使用，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。

主要内容

1. 双重加密及其安全性分析
2. 三重加密及其安全性分析

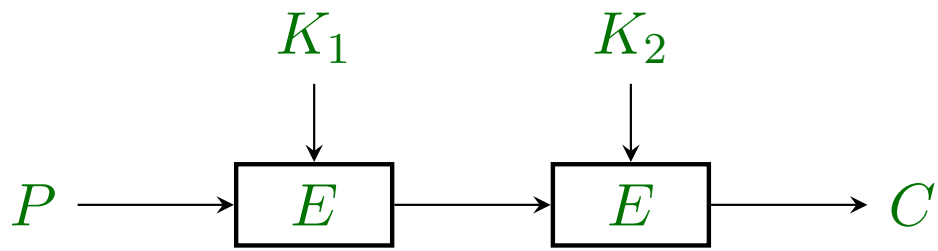
主要内容

1. 双重加密及其安全性分析

2. 三重加密及其安全性分析

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：$C = E_{K_2}(E_{K_1}(P))$

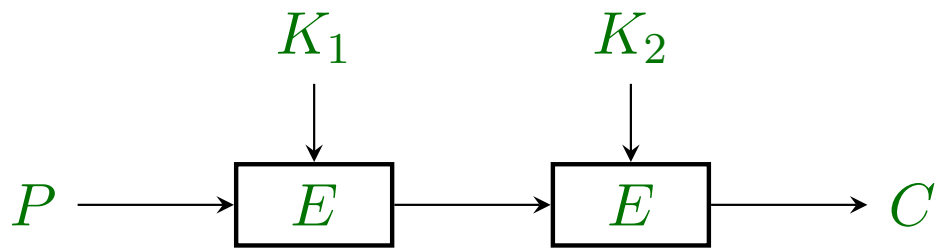


  - 解密：$P = D_{K_1}(D_{K_2}(C))$

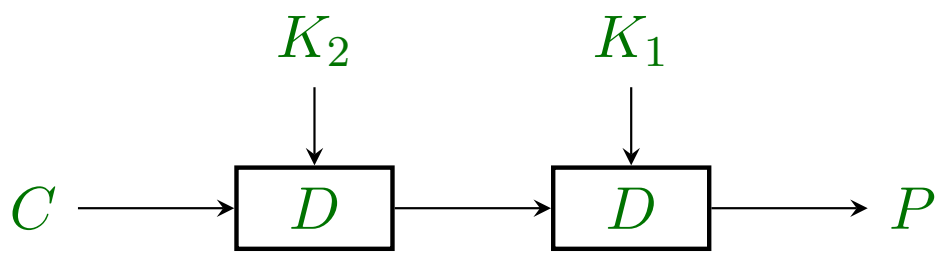
- 对于 DES，使用双重加密的 2DES 密钥长度为 112 位。

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：$C = E_{K_2}(E_{K_1}(P))$



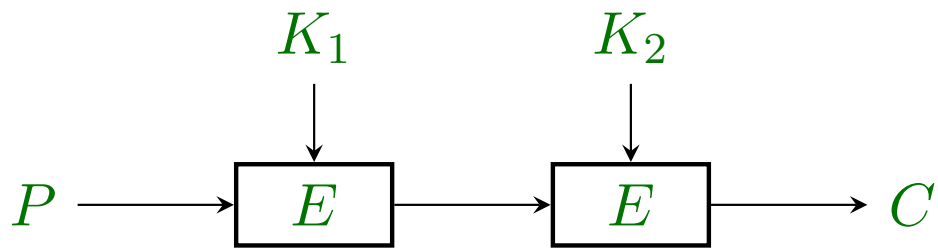
  - 解密：$P = D_{K_1}(D_{K_2}(C))$



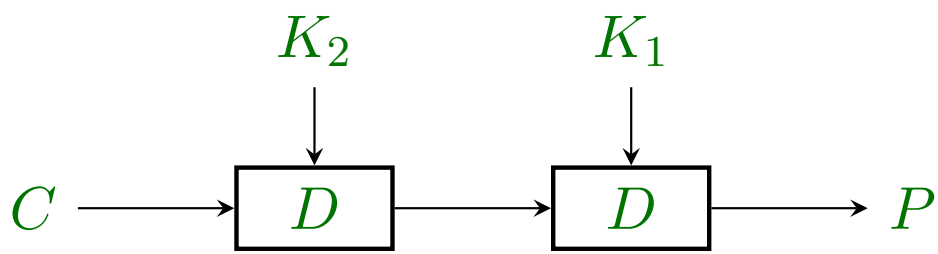
- 对于 DES，使用双重加密的 2DES 密钥长度为 112 位。

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：$C = E_{K_2}(E_{K_1}(P))$

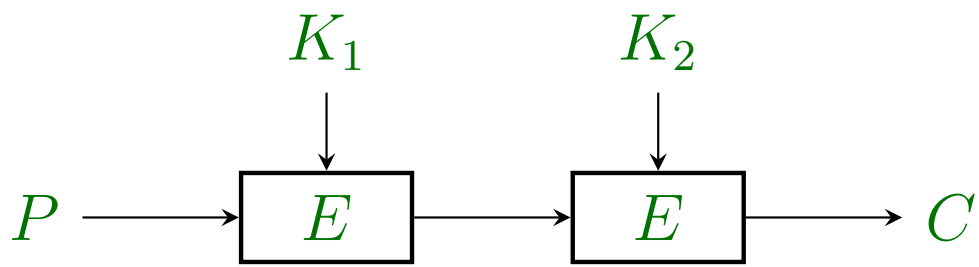


  - 解密：$P = D_{K_1}(D_{K_2}(C))$



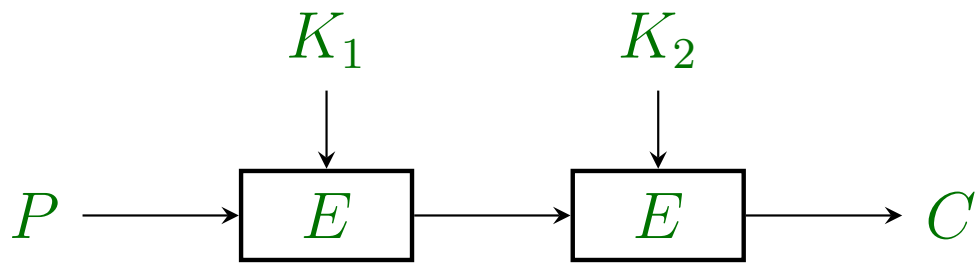
- 对于 DES，使用双重加密的 2DES 密钥长度为 112 位。

安全性分析：对 2DES 的穷举攻击（Brute-Force Attack）



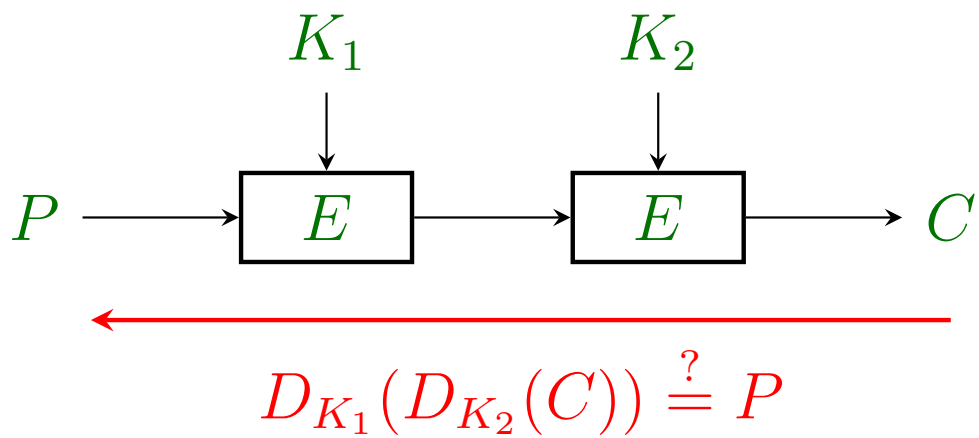
- 给定密文 C，依次尝试所有可能的密钥 K_2 和 K_1，直到发现明文 P
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次
- 有没有更有效的攻击手段？

安全性分析：对 2DES 的穷举攻击（Brute-Force Attack）



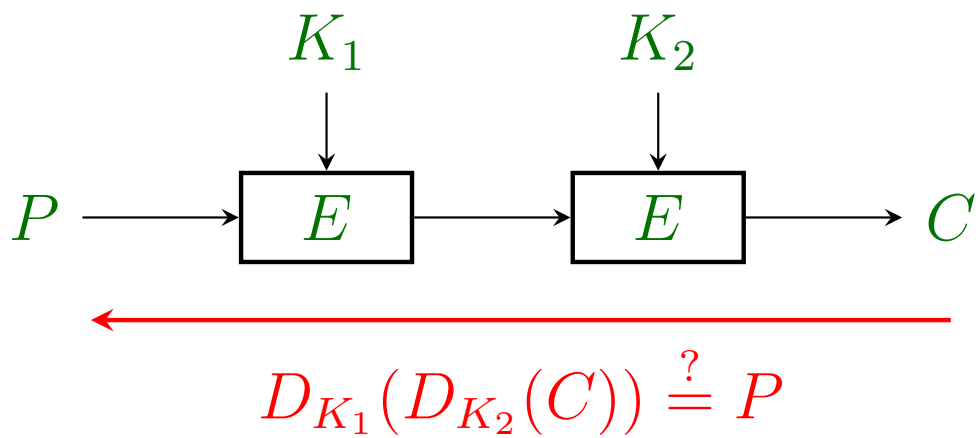
- 给定密文 C，依次尝试所有可能的密钥 K_2 和 K_1，直到发现明文 P
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次
- 有没有更有效的攻击手段？

安全性分析：对 2DES 的穷举攻击（Brute-Force Attack）



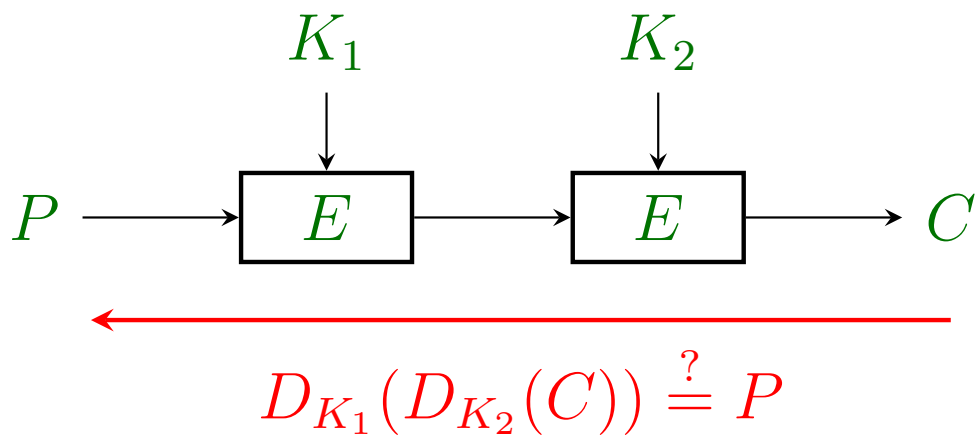
- 给定密文 C，依次尝试所有可能的密钥 K_2 和 K_1，直到发现明文 P
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次
- 有没有更有效的攻击手段？

安全性分析：对 2DES 的穷举攻击（Brute-Force Attack）



- 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 P
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次
- 有没有更有效的攻击手段？

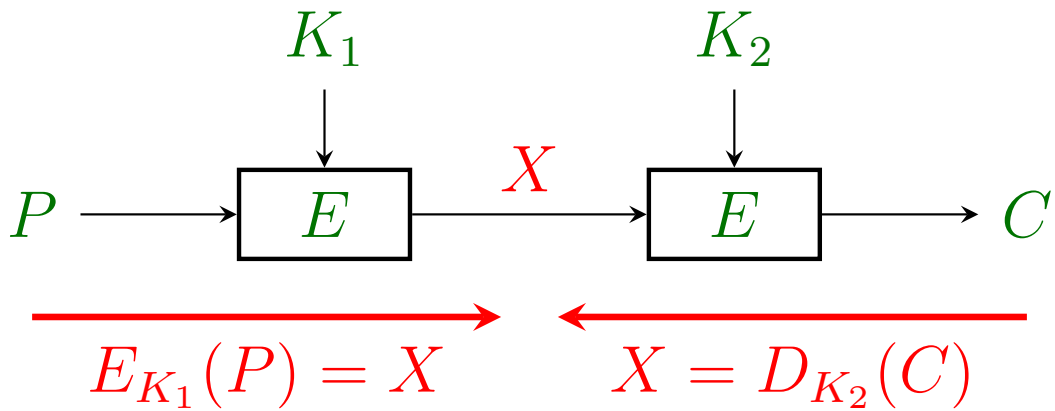
安全性分析：对 2DES 的穷举攻击 (Brute-Force Attack)



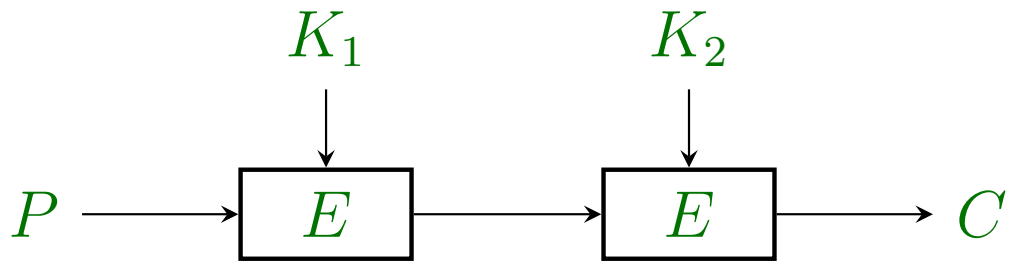
- 给定密文 C，依次尝试所有可能的密钥 K_2 和 K_1，直到发现明文 P
- 需要尝试 $2^{56} \times 2^{56} = 2^{112}$ 次
- 有没有更有效的攻击手段?

安全性分析：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。
- 最早在文献 Diffie, W., and Hellman, M. “Exhaustive Cryptanalysis of the NBS Data Encryption Standard.” *Computer*, June 1977 中提出。
- 基于如下观察：



攻击步骤

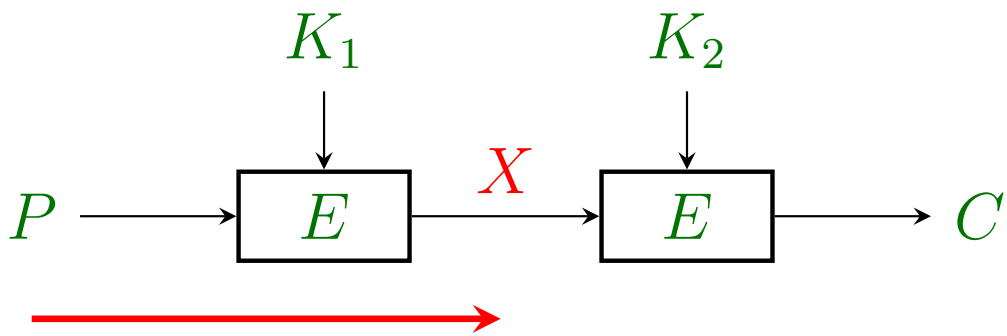


1. 将 P 按照所有可能的密钥 K_1 加密，得到的结果 X 排序后保存在一个表中；

2. 将 C 按照所有可能的密钥 K_2 解密，每解一次密，就将解密结果在表中匹配；

3. 如果产生匹配，说明得到一对可能密钥，然后用得到的两个密钥对一个新的明密文对进行验证，若通过则说明找到的密钥是正确密钥。

攻击步骤

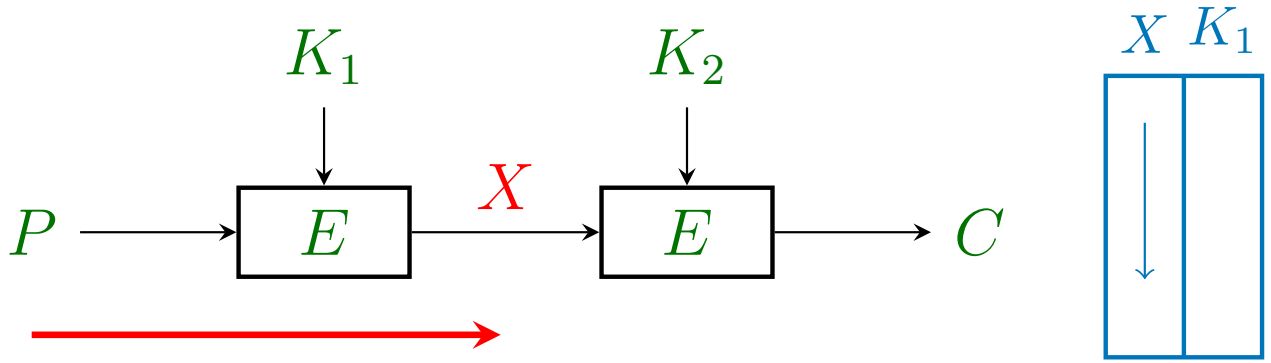


1. 将 P 按照所有可能的密钥 K_1 加密，得到的结果 X 排序后保存在一个表中；

2. 将 C 按照所有可能的密钥 K_2 解密，每解一次密，就将解密结果在表中匹配；

3. 如果产生匹配，说明得到一对可能密钥，然后用得到的两个密钥对一个新的明密文对进行验证，若通过则说明找到的密钥是正确密钥。

攻击步骤

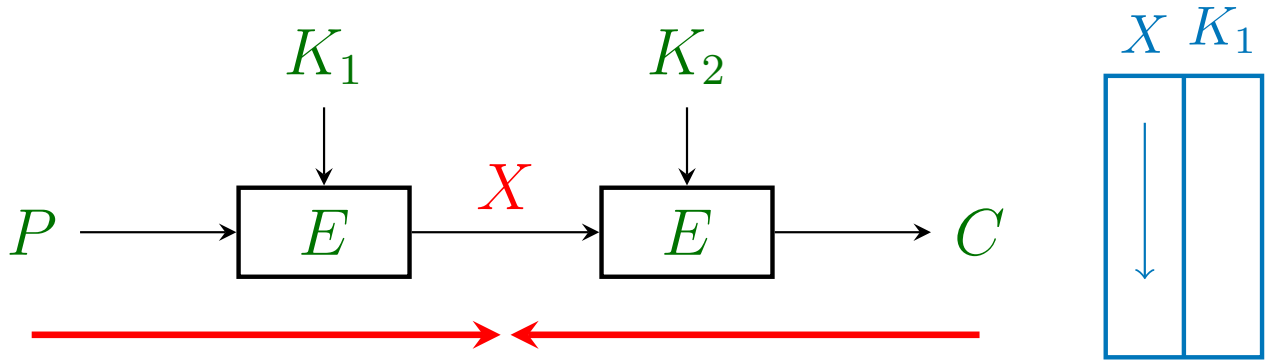


1. 将 P 按照所有可能的密钥 K_1 加密，得到的结果 X 排序后保存在一个表中；

2. 将 C 按照所有可能的密钥 K_2 解密，每解一次密，就将解密结果在表中匹配；

3. 如果产生匹配，说明得到一对可能密钥，然后用得到的两个密钥对一个新的明密文对进行验证，若通过则说明找到的密钥是正确密钥。

攻击步骤

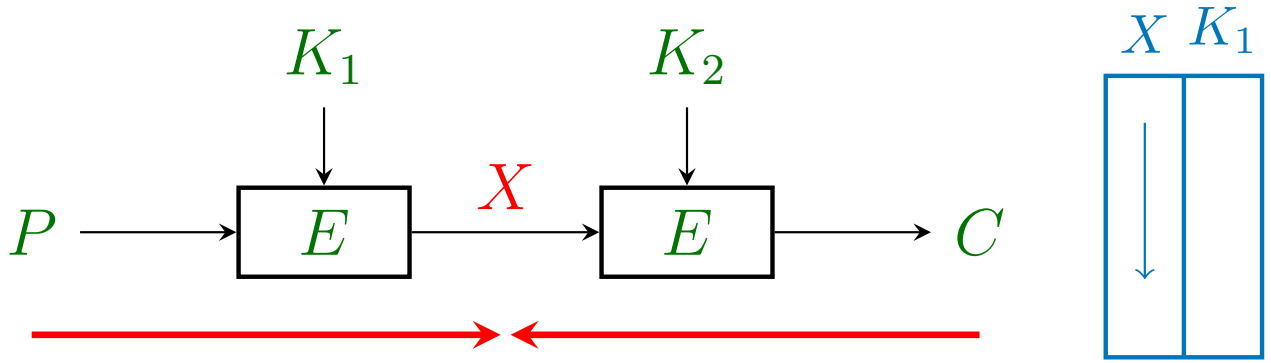


1. 将 P 按照所有可能的密钥 K_1 加密，得到的结果 X 排序后保存在一个表中；

2. 将 C 按照所有可能的密钥 K_2 解密，每解一次密，就将解密结果在表中匹配；

3. 如果产生匹配，说明得到一对可能密钥，然后用得到的两个密钥对一个新的明密文对进行验证，若通过则说明找到的密钥是正确密钥。

攻击步骤

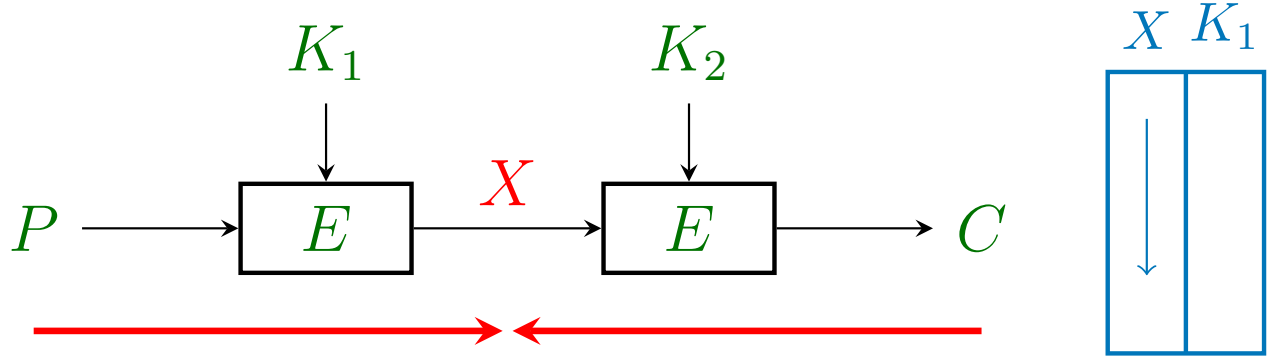


1. 将 P 按照所有可能的密钥 K_1 加密，得到的结果 X 排序后保存在一个表中；

2. 将 C 按照所有可能的密钥 K_2 解密，每解一次密，就将解密结果在表中匹配；

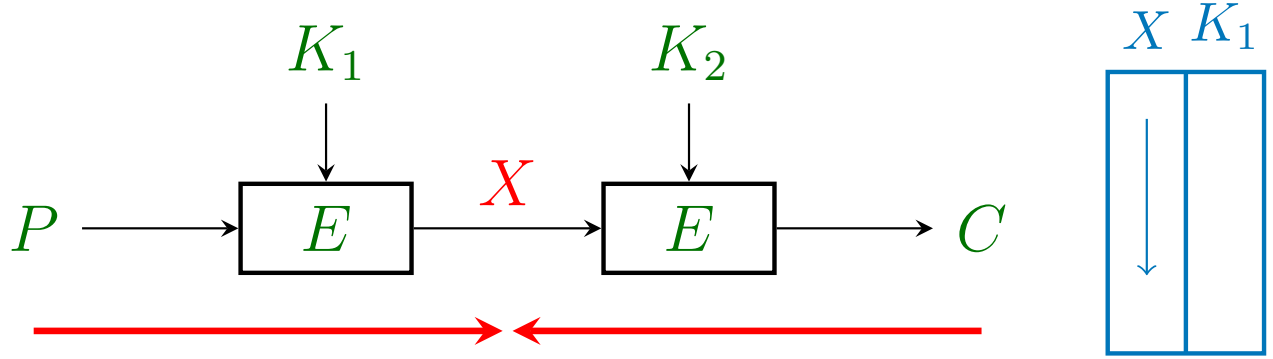
3. 如果产生匹配，说明得到一对可能密钥，然后用得到的两个密钥对一个新的明密文对进行验证，若通过则说明找到的密钥是正确密钥。

对 2DES 的中间相遇攻击复杂度分析



- 时间复杂度：$2^{56} + 2^{56} = 2^{57}$
- 空间复杂度：$(56 + 64) \times 2^{56}$ bit
- 第 3 步：
  - 使用一对 (P, C) 找到的错误密钥平均个数为：$2^{112}/2^{64} = 2^{48}$；
  - 使用两对 (P, C) 找到的错误密钥平均个数为：$2^{48}/2^{64} = 2^{-16} \approx 0$。
- 结论：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。

对 2DES 的中间相遇攻击复杂度分析



- 时间复杂度：$2^{56} + 2^{56} = 2^{57}$
- 空间复杂度：$(56 + 64) \times 2^{56}$ bit
- 第 3 步：
  - 使用一对 (P, C) 找到的错误密钥平均个数为：$2^{112}/2^{64} = 2^{48}$；
  - 使用两对 (P, C) 找到的错误密钥平均个数为：$2^{48}/2^{64} = 2^{-16} \approx 0$。
- 结论：**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。**

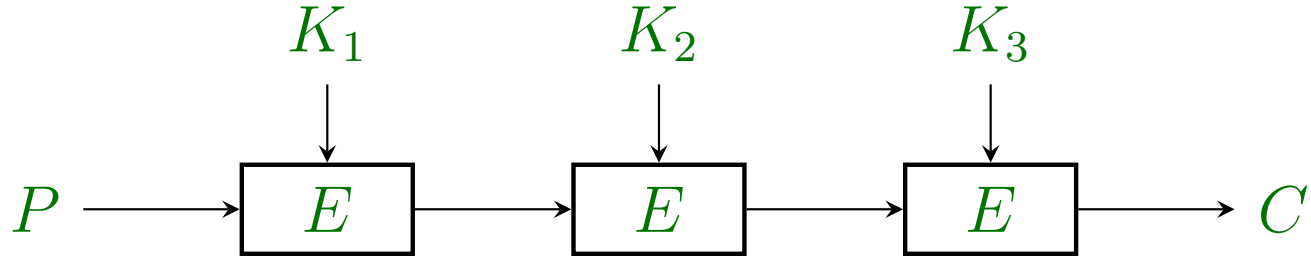
主要内容

1. 双重加密及其安全性分析

2. 三重加密及其安全性分析

三重加密与 3DES

- 为了对付中间相遇攻击，可以使用三重加密：

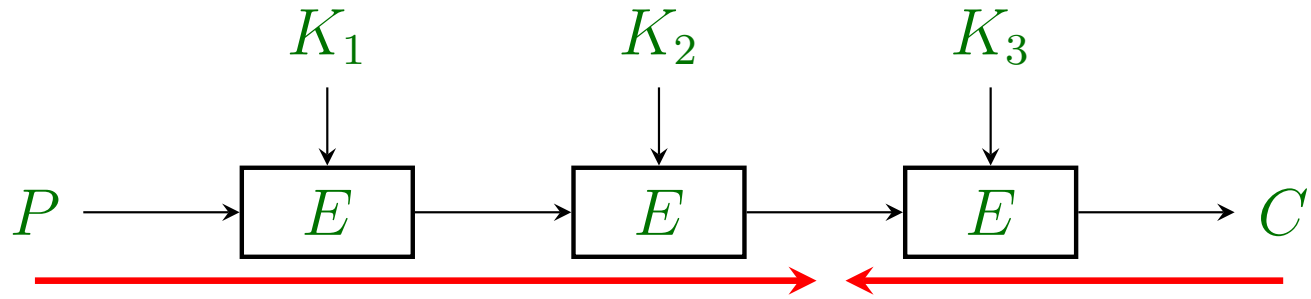


- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：

- 三种工作模式：
  - $K_1 \neq K_2 \neq K_3$ 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

三重加密与 3DES

- 为了对付中间相遇攻击，可以使用三重加密：

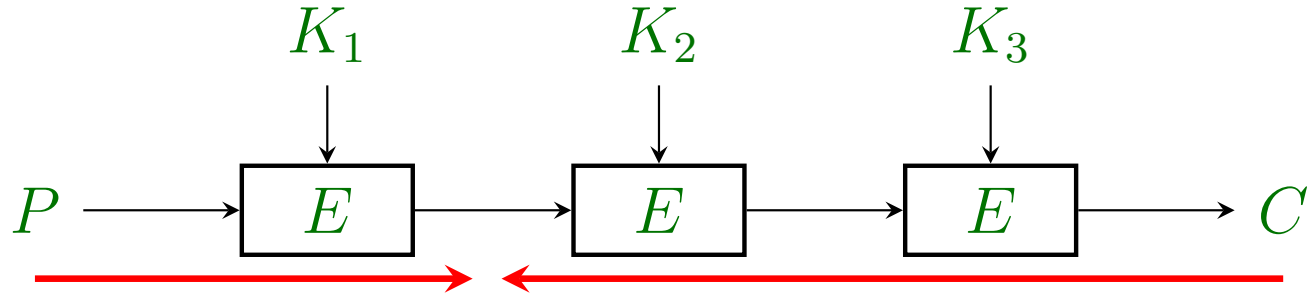


- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：

- 三种工作模式：
  - $K_1 \neq K_2 \neq K_3$ 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

三重加密与 3DES

- 为了对付中间相遇攻击，可以使用三重加密：

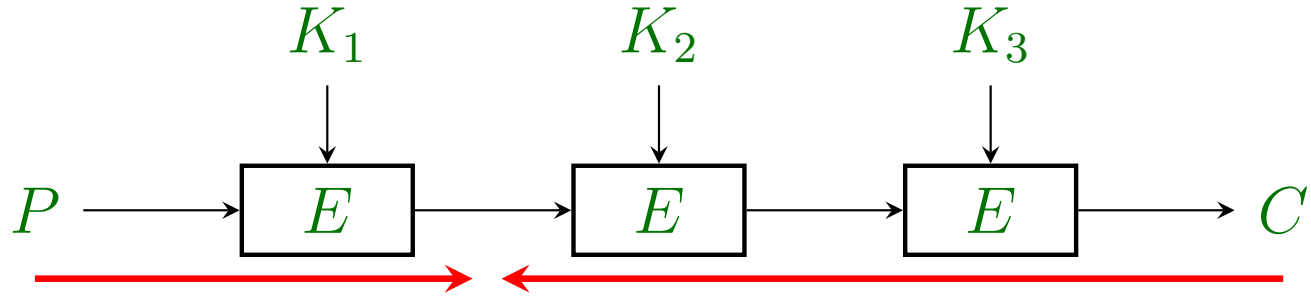


- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：

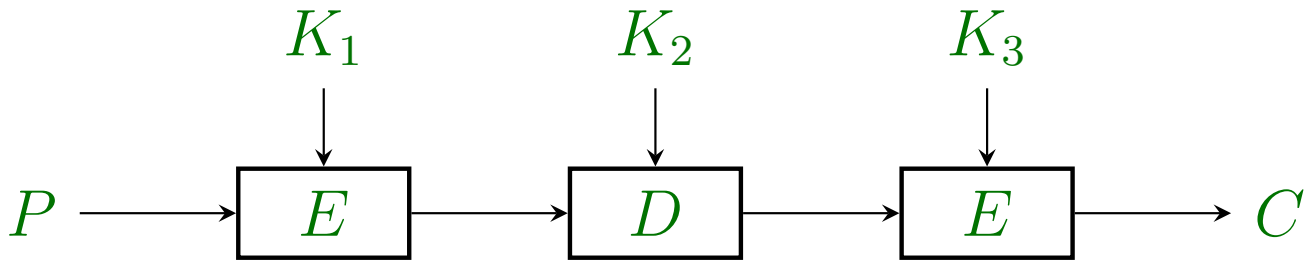
- 三种工作模式：
  - $K_1 \neq K_2 \neq K_3$ 强三重加密，密钥长度 $3 \times 56 = 168$
  - $K_1 = K_3 \neq K_2$ 普通三重加密，密钥长度 $2 \times 56 = 112$
  - $K_1 = K_2 = K_3$ 等价于普通分组加密，密钥长度 56

三重加密与 3DES

- 为了对付中间相遇攻击，可以使用三重加密：



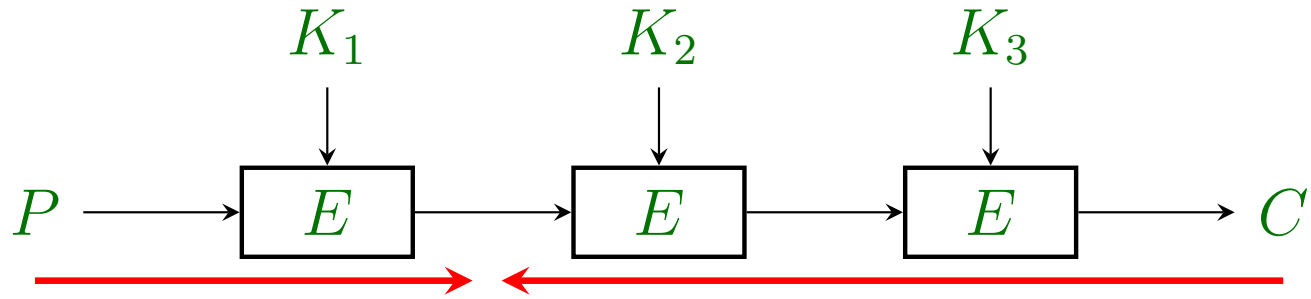
- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：



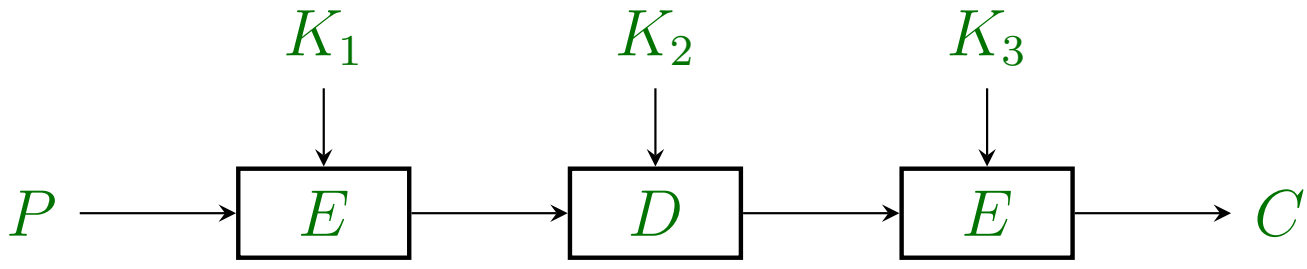
- 三种工作模式：
  - $K_1 \neq K_2 \neq K_3$ 强三重加密，密钥长度 $3 \times 56 = 168$
  - $K_1 = K_3 \neq K_2$ 普通三重加密，密钥长度 $2 \times 56 = 112$
  - $K_1 = K_2 = K_3$ 等价于普通分组加密，密钥长度 56

三重加密与 3DES

- 为了对付中间相遇攻击，可以使用三重加密：



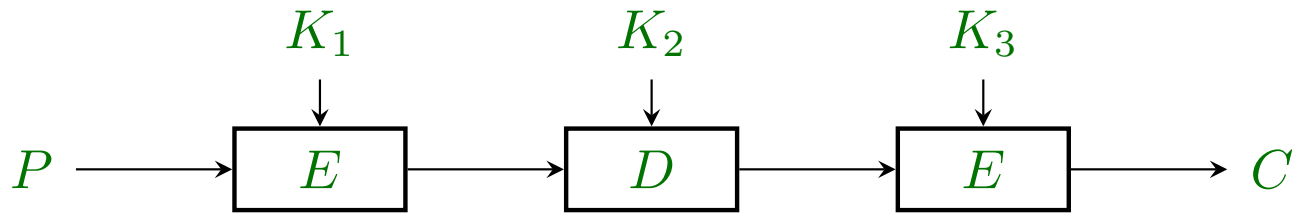
- 中间相遇攻击攻击的时间复杂度变为 $O(2^{112})$
- 实际中会使用如下更灵活的三重加密方案（RFC 1851）：



- 三种工作模式：
  - $K_1 \neq K_2 \neq K_3$ 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

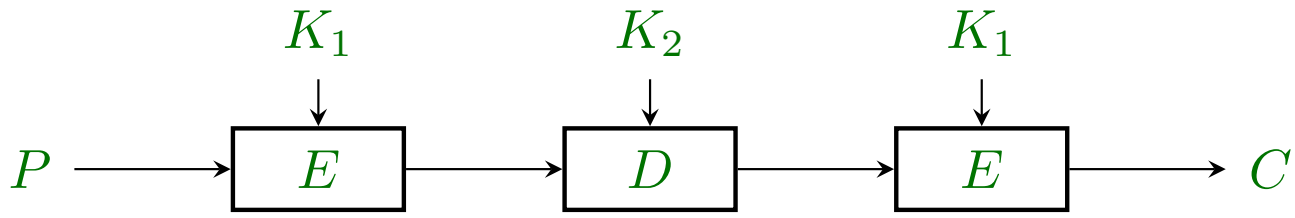
3DES 现状

- 使用工作模式 1 的 3DES 目前仍然被广泛应用；



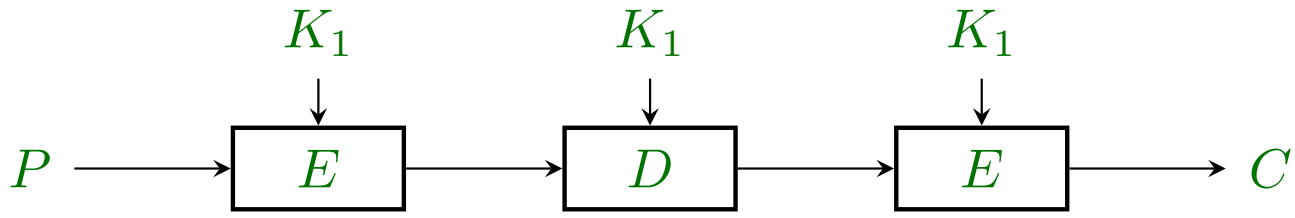
安全

- 使用工作模式 2 的 3DES 于 2017 年被认为已过时；



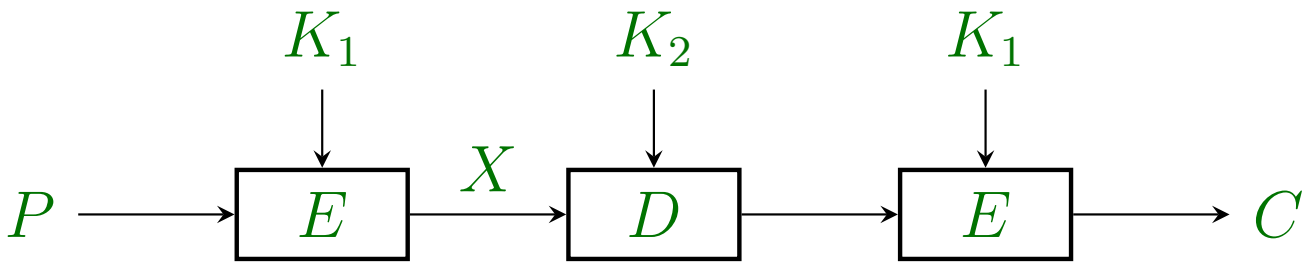
已过时

- 使用工作模式 3 的 3DES 等价于普通 DES，不安全。



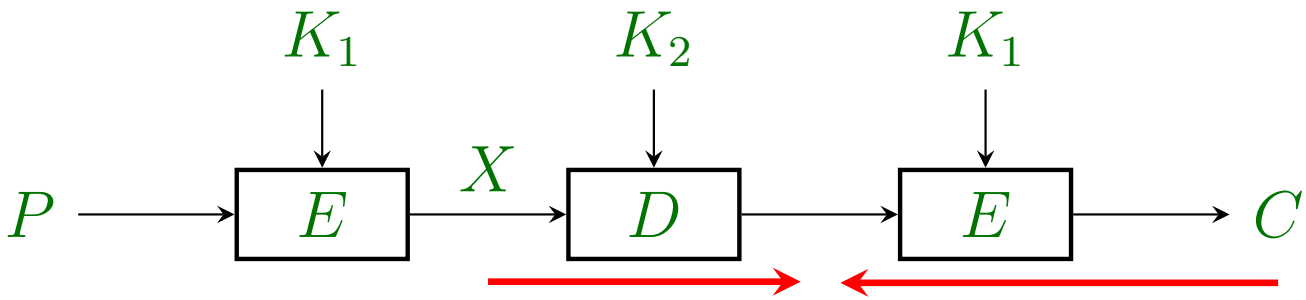
不安全

针对三重加密的已知明文攻击



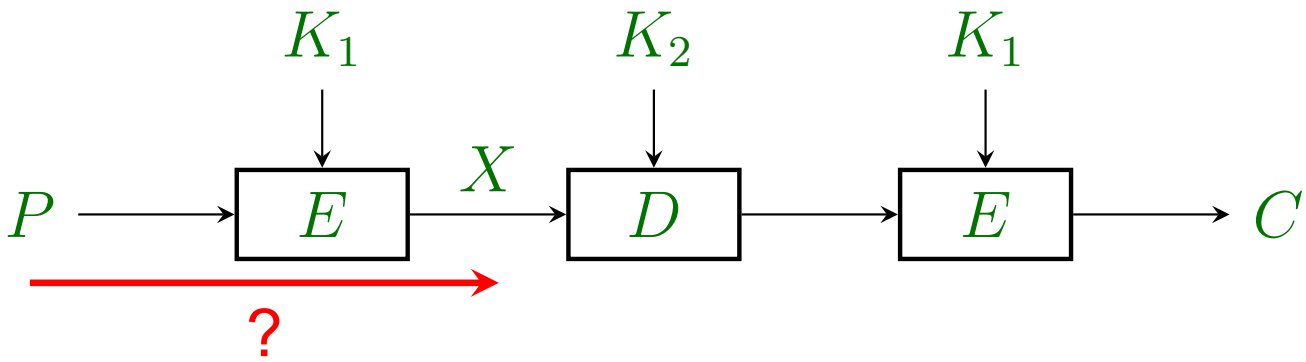
- 如果已知 X 和 C，那么对三重加密的攻击可以转化为对二重加密的攻击；
- 当然，只要不知道密钥，即使知道 P 和 C，还是无法知道 X；
- 然而，攻击者可以选择 X 的一个可能值，再试着找到一个可产生 X 的 (P, C) 对，从而将对三重加密的攻击转化为对二重加密的攻击。

针对三重加密的已知明文攻击



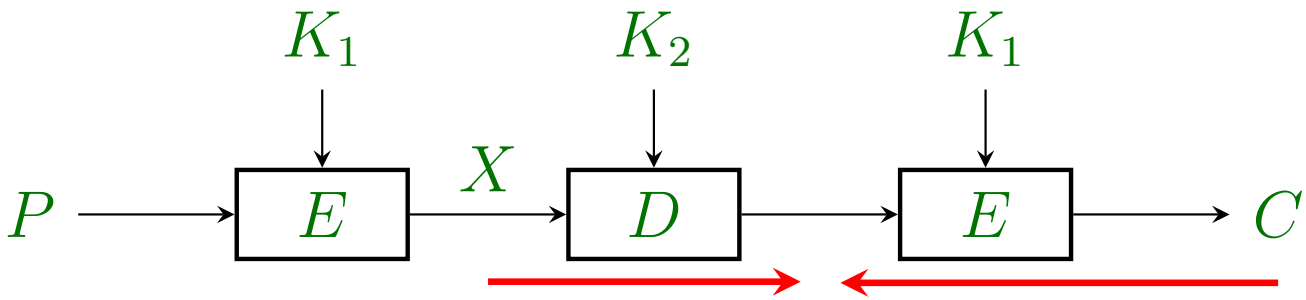
- 如果已知 X 和 C，那么对三重加密的攻击可以转化为对二重加密的攻击；
- 当然，只要不知道密钥，即使知道 P 和 C，还是无法知道 X；
- 然而，攻击者可以选择 X 的一个可能值，再试着找到一个可产生 X 的 (P, C) 对，从而将对三重加密的攻击转化为对二重加密的攻击。

针对三重加密的已知明文攻击



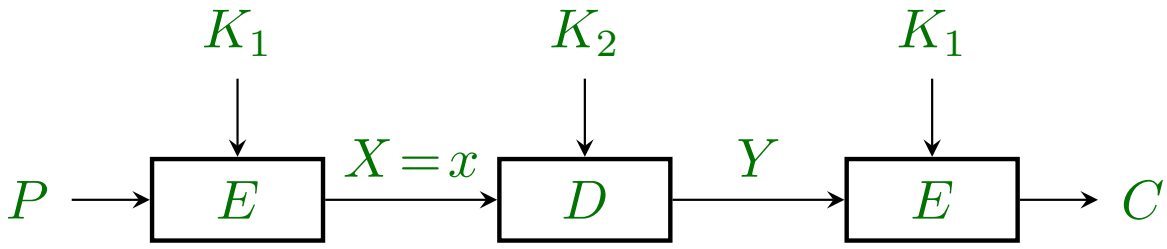
- 如果已知 X 和 C，那么对三重加密的攻击可以转化为对二重加密的攻击；
- 当然，只要不知道密钥，即使知道 P 和 C，还是无法知道 X；
- 然而，攻击者可以选择 X 的一个可能值，再试着找到一个可产生 X 的 (P, C) 对，从而将对三重加密的攻击转化为对二重加密的攻击。

针对三重加密的已知明文攻击



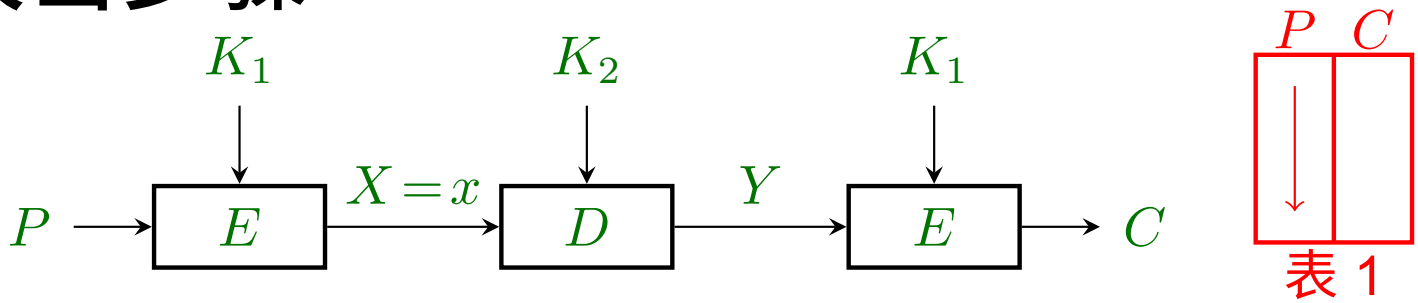
- 如果已知 X 和 C，那么对三重加密的攻击可以转化为对二重加密的攻击；
- 当然，只要不知道密钥，即使知道 P 和 C，还是无法知道 X；
- 然而，攻击者可以选择 X 的一个可能值，再试着找到一个可产生 X 的 (P, C) 对，从而将对三重加密的攻击转化为对二重加密的攻击。

攻击步骤



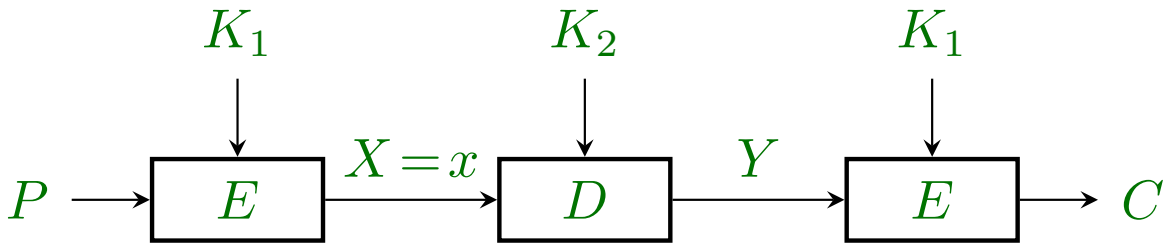
1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 随意选择值 x，按以下步骤创建表 2：
   - 对每个可能密钥 K_1 计算可产生 x 的明文 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
3. 搜索 K_2：
   - 对每个可能密钥 K_2，计算 $Y = D_{K_2}(x)$；
   - 在表 2 中匹配 Y，若匹配成功，则找到一对密钥 (K_1, K_2) 可以产生已知 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

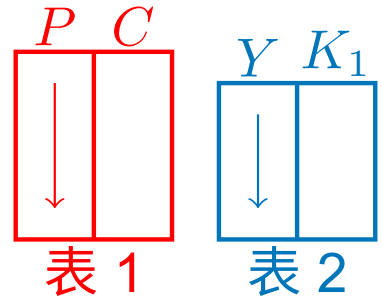
攻击步骤



1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 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 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
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 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

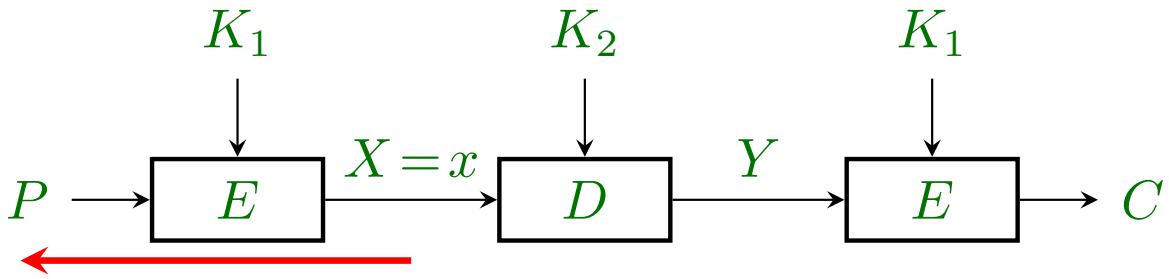
攻击步骤

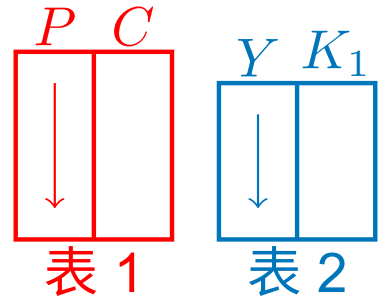




1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 随意选择值 x，按以下步骤创建表 2：
   - 对每个可能密钥 K_1 计算可产生 x 的明文 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
3. 搜索 K_2：
   - 对每个可能密钥 K_2，计算 $Y = D_{K_2}(x)$；
   - 在表 2 中匹配 Y，若匹配成功，则找到一对密钥 (K_1, K_2) 可以产生已知 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

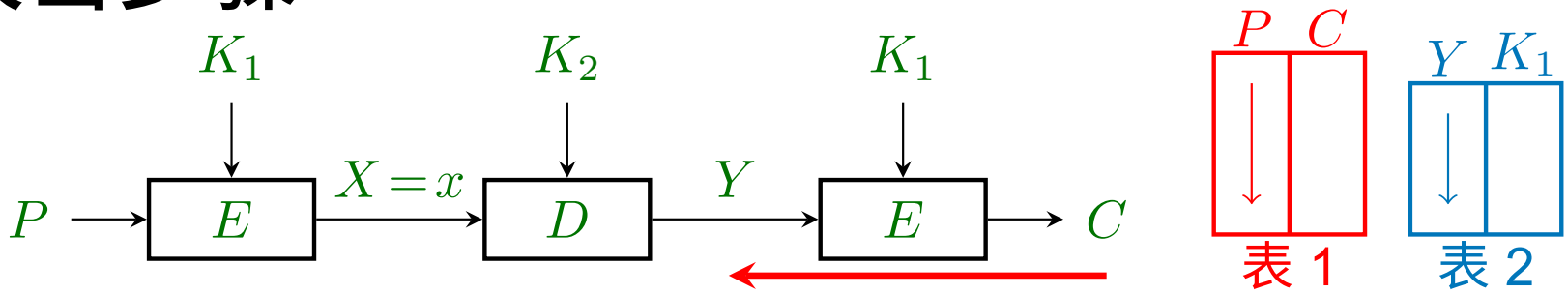
攻击步骤





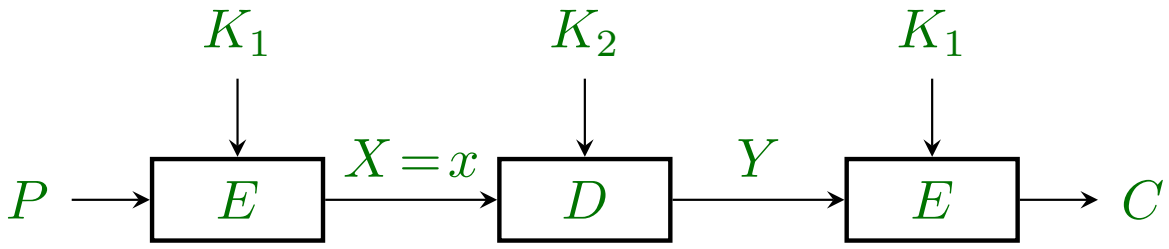
1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 随意选择值 x，按以下步骤创建表 2：
   - 对每个可能密钥 K_1 计算可产生 x 的明文 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
3. 搜索 K_2：
   - 对每个可能密钥 K_2，计算 $Y = D_{K_2}(x)$；
   - 在表 2 中匹配 Y，若匹配成功，则找到一对密钥 (K_1, K_2) 可以产生已知 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

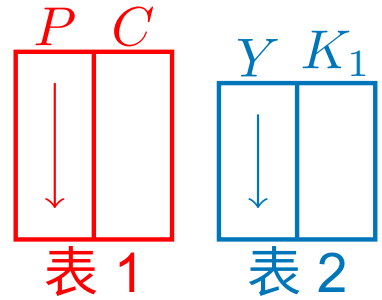
攻击步骤



1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 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 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
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 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

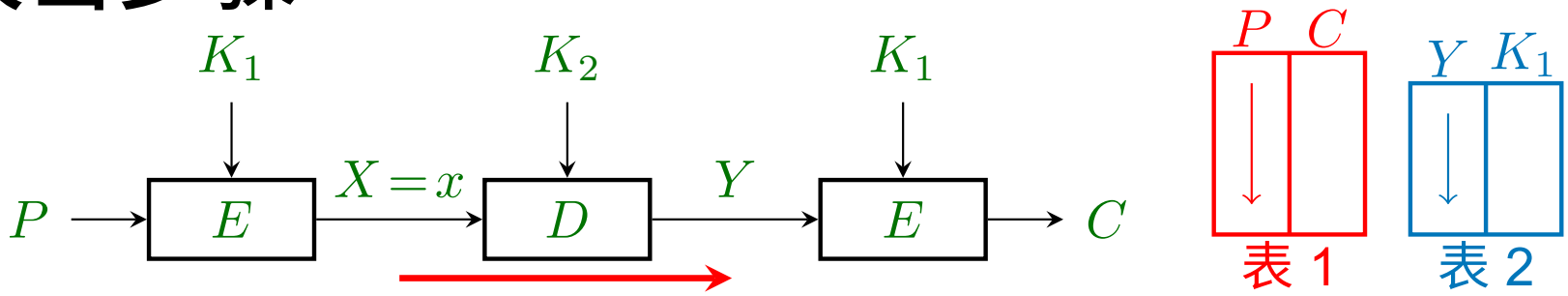
攻击步骤





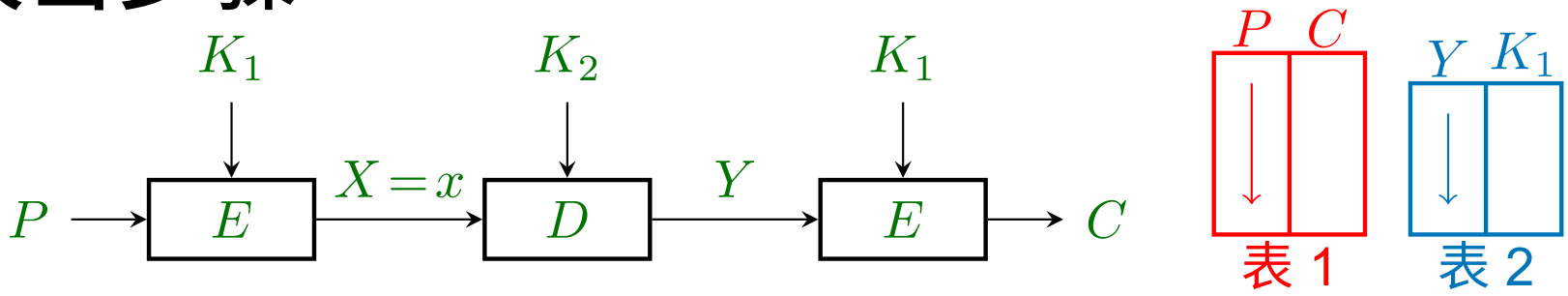
1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 随意选择值 x，按以下步骤创建表 2：
   - 对每个可能密钥 K_1 计算可产生 x 的明文 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
3. 搜索 K_2：
   - 对每个可能密钥 K_2，计算 $Y = D_{K_2}(x)$；
   - 在表 2 中匹配 Y，若匹配成功，则找到一对密钥 (K_1, K_2) 可以产生已知 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

攻击步骤



1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 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 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
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 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

攻击步骤



1. 获取尽量多个 (P, C) 对，按 P 排序后存入表 1 中；
2. 为 X 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 $P = D_{K_1}(x)$；
   - 在表 1 中匹配 P，若匹配成功，则在表 2 中添加一项 (Y, K_1)，其中 $Y = D_{K_1}(C)$，表 2 按 Y 排序。
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 (P, C) 对。
4. 对找到的密钥对在其他 (P, C) 对上验证，若验证失败则返回步骤 2；否则，表示找到正确密钥对 (K_1, K_2)。

复杂度分析

- 对给定的 (P, C) 对，选择 $X = x$ 成功的可能性为 2^{-64};
- 给定 n 个 (P, C) 对，则选择 $X = x$ 成功的可能性为 $2^{-64}n$;
- 所以，平均需要尝试 $2^{64}/n$ 次，才会得到一个正确的 x;
- 因此，总的时间复杂度为量级：$2^{57}2^{64}/n = 2^{121-\log_2 n}$。

小结

1. 双重加密及其安全性分析
2. 三重加密及其安全性分析