密码学理论第三次作业

朱哲昊

March 2023

1 理想分组密码

(1) 对于 n 位分组长度的理想分组密码, 明文空间为 2^n，密文空间为 2^n, 从明文到密文的可逆映射需要保证明文到密文的一一映射，故可逆映射有 $2^n!$ 个

(2) 明文有 2^n 个，密文也有 2^n 个，则对应的映射，也就是密钥个数为 2^{n2^n} 个，但是这中间的密钥一定存在相同的情况，那么我们可以知道密钥空间的大小实际上是小于这个值的

(3) 从 1 到 $2^n!$, 若有一个表为 $\log_2 2^n!$ 长，每一位用 0，1 表示，则可以表示 $2^n!$ 种映射方式，即也能表示对应的 $2^n!$ 种唯一的映射

(4) 用 $n * 2^n$ 表示:

明文	密文
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111

表 1: 明文-密文

密钥为: 1110, 0100, 1101, 0001, 0010, 1111, 1011, 1000, 0011, 1010, 0110, 1100, 0101, 1001, 0000, 0111
用 $\log_2 m!$ 表示

$3*9!+(6-1)*8!+(5-1)*7!+(8-3)*6!+(7-3)*5!+1*4!+(4-2)*3!+(9-7)*2!+0=1314520$

表示成二进制就是 0001, 0100, 0000, 1110, 1101, 1000

2 Feistel 结构

(1) 解密过程, 轮密钥就是加密时的轮密钥从第 15 个到第 0 个的使用, 我们观察加密的第 16 轮和解密的第 1 轮。
对于加密而言有 $LE16 = RE15\ RE16 = LE15 \oplus f(RE15, K16)$, 对解密而言有 $LD1 = RD0 = LE16 = RE15, RD1 = LD0 \oplus f(RD0, K16) = RE16 \oplus f(LE16, K16) = (LE15 \oplus f(RE15, K16)) \oplus f(RE15, K16) = LE15$ 故有:

$$LD1 = RE15, RD1 = LE15$$

由此可以得出, 解密算法是加密算法的逆

(2) 在密钥 k 满足这样的情况下，加密和解密可以互相替代，对得到的密文 c 进行加密 Enc，也就会得到我们想要的明文 m

(3) 对于 i 来说，$L_{n+2} = R_{n+1} = L_n \oplus 1 = L'_n$,$L_16 = L0, R_16 = R_0$, 则密文 $R_16L_16 = R_0L_0$, 输出的密文就是 $R_17L_17 = L_0R_0$ 相当于明文的左右部分互换，加密效果很差
对 ii 来说，$R_{n+2} = L_{n+1} \oplus R'_{n+1} = R_n \oplus (L_n \oplus R'_n)' = L_n$ 有 $L + n + 3 = R_{n+2} = L_n$ 由此可以得出 $L_16 = L_1 = R_0, R_16 = R_1 = L_0 \oplus R'_0$ 密文 $L_17R_17 = L_0 \oplus R'_0R_0$, 密文也非常好破解，安全性不佳
此时加密相当将明文左半部分 L 与右半部分 R 的逆异或加上 R。对于解密，密文左半部分与右半部分异或的结果的逆和密文的右半部分合起来即为明文。

3 扩散和混淆

(1) 扩散是指把明文的统计特征消散在密文中，让每个明文数字尽可能多的影响密文数字
混淆是指尽可能使明文和加密密钥之间的关系更加复杂
扩散是对明文进行置换等操作让其统计规律消散，而混淆是明文和密文进行一定的运算

(2) 古典密码的代替与置换一般为$_{A\quad\quad C\quad\quad F\ D\quad\quad E\ A.\ B.\ C.\ D.\ E.\ F.}$

(3) IP 置换和 IP 逆置换是扩散，扩展置换是扩散，与密钥的异或运算是混淆，S 盒代替是混淆，P 盒代替是扩散，密文中间两部分异或运算是扩散

(4) 构造 N 个长为 N 的字符串 $msg_i = 00....010....00$(第 i 位为 1, i<=N), 他, 对应的密文为 $cipher_i$, 对于我们想要加密的明文，找出各个 1 的对应位，将对应的 $cipher_i$ 进行异或运算，所得到的就是我们要的密文串

4 DES 算法

(1) 密钥输入为 64 位，密钥空间为 2^56 位, 因为在 64 位中有 8 位的奇偶校验位，实际上进行应用的只有 56 位

(2) 密钥取反，经过子密钥生成运算，结果也是原本的子密钥取反，再加上明文也是取反的, 由异或的性质，即 $A \oplus B = A' \oplus B'$，则第一轮加密，中间的密文没有变化，第二轮加密时，由于异或的性质 $A \oplus B' = (A \oplus B)'$, 而一共有 16 轮加密则最终的密文为原本的密文取反
每一次对随机取得密钥进行判断的时候，只需另再判断明文取反和密钥取反的加密结果，也能达到判断该取反式的效果，故搜索空间为 2^{55}

(3) 我们详细观察轮密钥的生成过程，我们发现密钥经过一次置换之后被分成 C_0 和 D_0 来进行运算，每一轮的密钥生成就是对 C 和 D 进行左移，之后拼接在一起再进行运算。每个 C 和 D 都有 28 位，经过置换之后就是 24 位，而这 24 位都是来自于对应的 28 位的 C 或 D，也就来自对应的初始密钥的前 28 位或后 28 位

5 DES 算法

(1) 第一轮密钥为 0xb02679b49a5

(2) $l_0 = 1100, 1100, 0000, 0000, 1100, 1100, 1111, 1111$ $r_0 = 1111, 0000, 1010, 1010, 1111, 0000, 1010, 1010$

(3) $E[r_0] = 0111, 1010, 0001, 0101, 0101, 0101, 0111, 1010, 0001, 0101, 0101, 0101$

(4) $A = E[r_0] \oplus K1 = 0111, 0001, 0001, 0111, 0011, 0010, 1110, 0001, 0101, 1100, 1111, 0000$

(5) 分成 6*8 的形式：$011100, 010001, 011100, 110010, 111000, 010101, 110011, 110000$ $valueS1 = 0000$
$valueS2 = 1100$
$valueS3 = 0010$
$valueS4 = 0001$
$valueS5 = 0110$
$valueS6 = 1101$
$valueS7 = 0101$
$valueS8 = 0000$

(6) $B = 0000, 1100, 0010, 0001, 0110, 1101, 0101, 0000$

(7) $P(B) = 1001, 0010, 0001, 1100, 0010, 0000, 1001, 1100$

(8) $P(B) \oplus L_0 = 0101, 1110, 0001, 1100, 1110, 1100, 0110, 0011$

(9) 密文为 $0x56cc09e7cfdc4cef$