RC4加密算法

Rivest Cipher4

2021·08·29

基本原理

RC4属于对称密码算法中的流密码加密算法。

密钥长度可变，面向字节操作。

它以一个足够大的表S为基础，对表进行非线性变换，产生密钥流。

明文 ⊕ 密钥 ⟶ 密文

密文 ⊕ 密钥 ⟶ 明文

加密过程

一、初始化S表

Step1:对S表进行线性填充，一般为256个字节；

Step2:用种子密钥填充另一个256字节的K表；

Step3:用K表对S表进行初始置换。

二、密钥流的生成（为每个待加密的字节生成一个伪随机数，用来异或）

注：表S一旦完成初始化，种子密钥就不再被使用。

一、初始化S表

Step1:对S表进行线性填充，一般为256个字节；

Step2:用种子密钥填充另一个256字节的K表；

Step3:用K表对S表进行初始置换。

S 表	0	1	2	3	4	5	6
	S[0]	S[1]	S[2]	S[3]	S[4]	S[5]	S[6]

K 表	3	4	5	3	4	5	3
	K[0]	K[1]	K[2]	K[3]	K[4]	K[5]	K[6]

```
j=0;
for i=0   to 255   do
    j= (j + S [i] + K [i]) mod 256;
    Swap (S [i], S [j]);
```

S 表	3	0	1	4	5	2	6
	S[0]	S[1]	S[2]	S[3]	S[4]	S[5]	S[6]

二、密钥流的生成（为每个待加密的字节生成一个伪随机数，用来异或）。

```
i,j=0;
for r=0 to len do  //r为明文长度，r字节
   i=(i+1) mod 256;
   j=(j+S[i])mod 256;
   swap(S[i],S[j]);
   t=(S[i]+S[j])mod 256;
   k[r]=S[t];
```

S 表	3	0	1	4	5	2	6
	S[0]	S[1]	S[2]	S[3]	S[4]	S[5]	S[6]

$$i = (i+1) \bmod 7 = 0+1=1;$$

$$j = (j+S[i]) \bmod 7 = 0+S[1] = 0+0=0;$$

Swap(S[0], S[1])

$$t = S[0]+S[1] \bmod 7 = 3;$$

$$S[3] = 4;$$

$$K[0] = S[3] = 4$$