第4章 分组密码

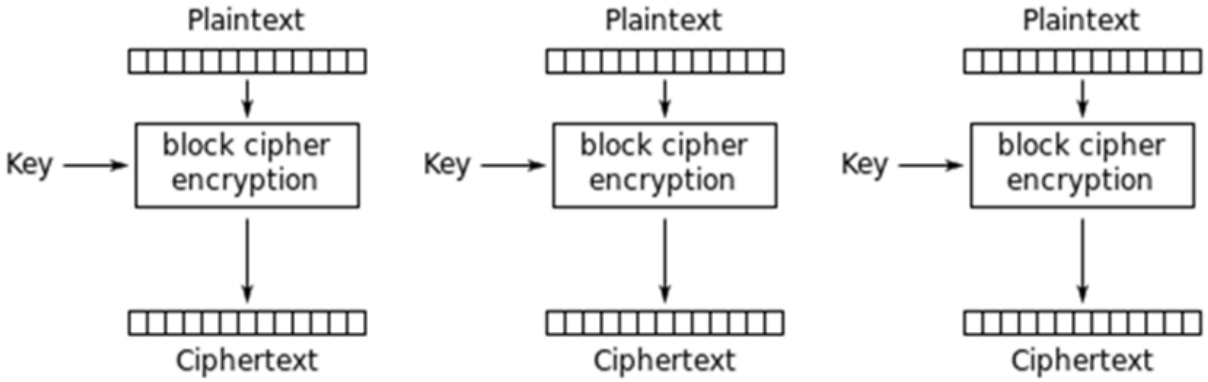
分组密码工作模式

分块加密，每块8字节

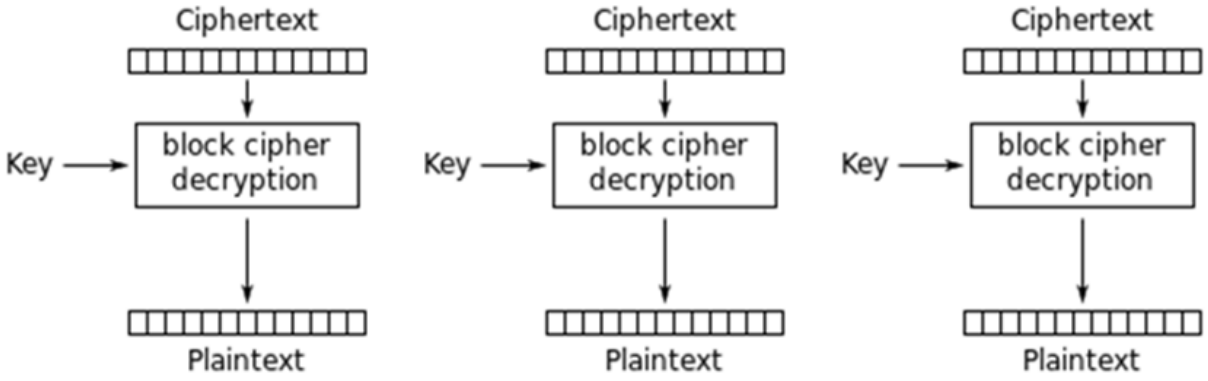
1、电子密码簿ECB

$$\begin{cases} \text{加密：} C_j = E_k(P_j) \\ \text{解密：} P_j = D_k(C_j) \end{cases}$$

- 效率高，加密解密都可以并行计算
- 安全性较差，对于相同内容的明文段，加密后得到的密文块是相同的



Electronic Codebook (ECB) mode encryption



Electronic Codebook (ECB) mode decryption

2、密码块链接模式CBC(Cypher Block Chaining)

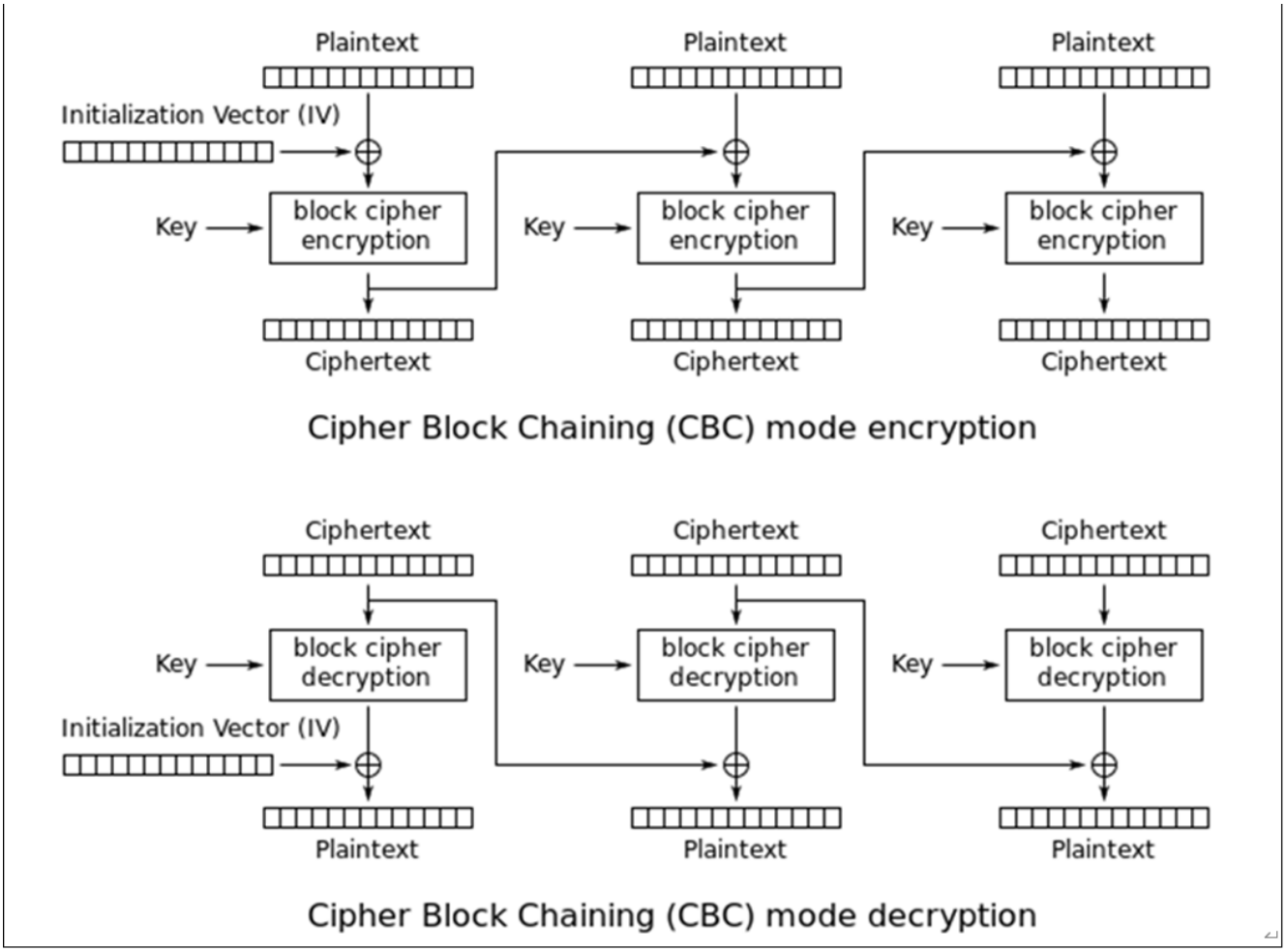
对不同的block建立关联

$$\begin{cases} 加密：C_j = E_k(P_j \bigoplus C_{j-1}) \\ 解密：P_j = D_k(C_j) \bigoplus C_{j-1} \end{cases}$$

- 加密过程
  - Initializing Vector(IV) 种子 和明文长度一致
  - $IV \bigoplus Plaintext$
  - 对结果用加密算法进行计算，得到密文

> 不能并行计算不同的块，没有上一块的值就不能进行下一块的计算

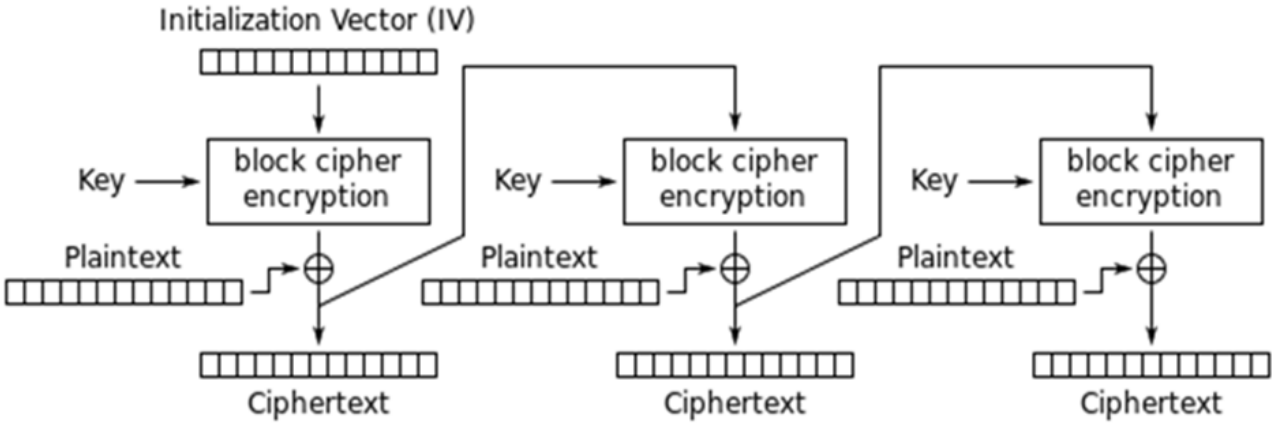
- 解密过程
  - 对密文用解密函数解密
  - 结果与IV进行异或运算得明文



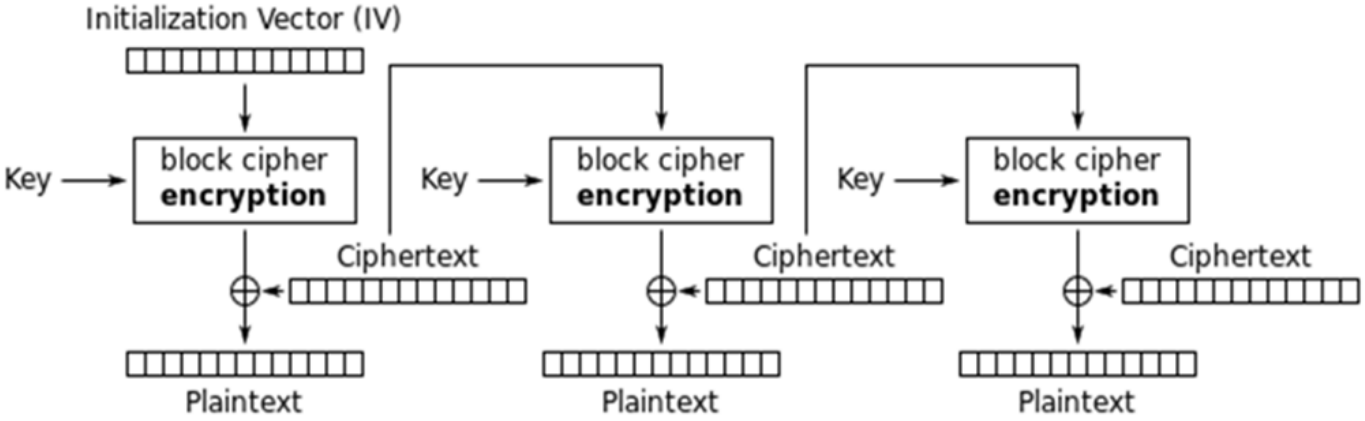
3、密文反馈模式CFB(Cypher Feedback)

- 把种子（IV）当作明文用加密函数加密，
- 结果再和明文进行异或运算得到密文

```
# Round I
Plaintext:p[0] p[1] p[2] p[3] p[4] p[5] p[6] p[7]
IV: iv[0] iv[1] iv[2] iv[3] iv[4] iv[5] iv[6] iv[7]
Enctrpty IV: iv'[0] iv'[1] iv'[2] iv'[3] iv'[4] iv'[5] iv'[6] iv'[7]
Cyphertext:c[0] * * * * * * * #c[0]=iv'[0]⊕p[0]
# Round II
Plaintext:p[0] p[1] p[2] p[3] p[4] p[5] p[6] p[7]
IV: iv[1] iv[2] iv[3] iv[4] iv[5] iv[6] iv[7] c[0]
Enctrpty IV: iv'[1] iv'[2] iv'[3] iv'[4] iv'[5] iv'[6] iv'[7] c'[0]
Cyphertext:c[0] c[1] * * * * * * #c[0]=iv'[0]⊕p[0] c[1]=iv'[1]⊕p[1]
...
Cyphertext:c[0] c[1] c[2] c[3] c[4] c[5] c[6] c[7]
```



Cipher Feedback (CFB) mode encryption



Cipher Feedback (CFB) mode decryption

> 可以从密文传输的错误中恢复

RC4流密码

逐字节加密 每次循环均加密一个字节
计算过程简单，算法效率高
加密函数和解密函数相同

```
for(counter=0;counter<256;counter++)
    state[counter]=counter;

buffer_ptr[count]^=state[xorindex];
```