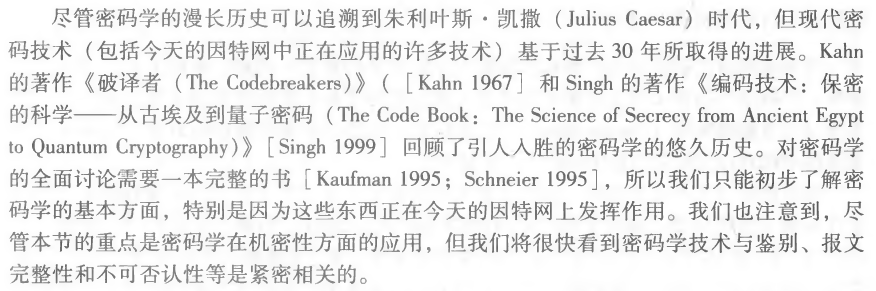
# 密码学：cryptography是什么

## 参1：《计算机网络 谢希仁》v7



# 密码学的原则/数据加密模型+密码学常用术语

## 参1：

<https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6>

直到现代以前，密码学几乎专指[**加密**](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)**算法**：将普通信息（[**明文**](https://zh.wikipedia.org/wiki/%E6%98%8E%E6%96%87)）转换成难以理解的资料（[**密文**](https://zh.wikipedia.org/wiki/%E5%AF%86%E6%96%87)）的过程；[**解密**](https://zh.wikipedia.org/wiki/%E8%A7%A3%E5%AF%86)**算法**则是其相反的过程：由[密文](https://zh.wikipedia.org/wiki/%E5%AF%86%E6%96%87)转换回[明文](https://zh.wikipedia.org/wiki/%E6%98%8E%E6%96%87)；加解密包含了这两种[算法](https://zh.wikipedia.org/wiki/%E7%AE%97%E6%B3%95)，一般[加密](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)即同时指称[加密](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)与[解密](https://zh.wikipedia.org/wiki/%E8%A7%A3%E5%AF%86)的技术。

加解密的具体运作由两部分决定：一个是[**算法**](https://zh.wikipedia.org/wiki/%E7%AE%97%E6%B3%95)，另一个是[**密钥**](https://zh.wikipedia.org/wiki/%E5%AF%86%E9%92%A5)。[密钥](https://zh.wikipedia.org/wiki/%E5%AF%86%E9%92%A5)是一个用于加解密算法的秘密参数，通常只有通信者拥有。历史上，[密钥](https://zh.wikipedia.org/wiki/%E5%AF%86%E9%92%A5)通常未经[认证](https://zh.wikipedia.org/wiki/%E8%AA%8D%E8%AD%89)或[完整性](https://zh.wikipedia.org/wiki/%E5%AE%8C%E6%95%B4%E6%80%A7)测试而被直接使用在[加解密](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E6%9C%BA)上。

。。。。。。

## 参2：

<https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001>

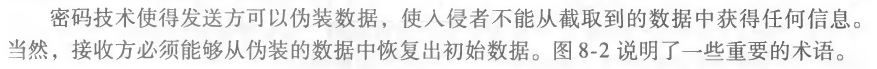
直到现代以前，密码学几乎专指[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86)(encryption）算法：将普通信息（[明文](https://baike.baidu.com/item/%E6%98%8E%E6%96%87),plaintext）转换成难以理解的资料（[密文](https://baike.baidu.com/item/%E5%AF%86%E6%96%87),ciphertext）的过程；[解密](https://baike.baidu.com/item/%E8%A7%A3%E5%AF%86)(decryption）算法则是其相反的过程：由[密文](https://baike.baidu.com/item/%E5%AF%86%E6%96%87)转换回[明文](https://baike.baidu.com/item/%E6%98%8E%E6%96%87)；加解密包含了这两种[算法](https://baike.baidu.com/item/%E7%AE%97%E6%B3%95)，一般[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86)即同时指称[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86)(encrypt或encipher）与[解密](https://baike.baidu.com/item/%E8%A7%A3%E5%AF%86)(decrypt或decipher）的技术。

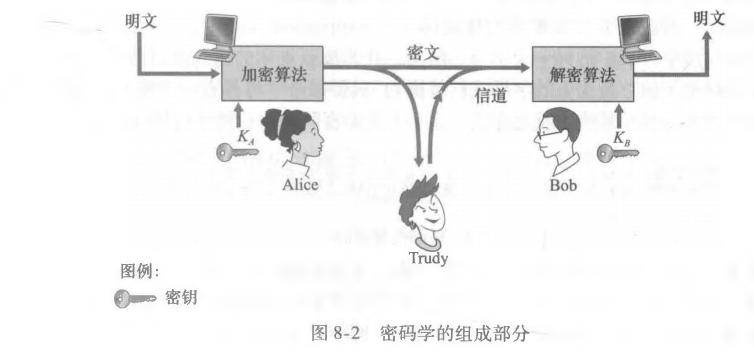
加解密的具体运作由两部分决定：一个是[算法](https://baike.baidu.com/item/%E7%AE%97%E6%B3%95)，另一个是[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5)。[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5)是一个用于加解密算法的秘密参数，通常只有通讯者拥有。历史上，[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5)通常未经[认证](https://baike.baidu.com/item/%E8%AE%A4%E8%AF%81)或[完整性](https://baike.baidu.com/item/%E5%AE%8C%E6%95%B4%E6%80%A7)测试而被直接使用在[密码机](https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E6%9C%BA)上。

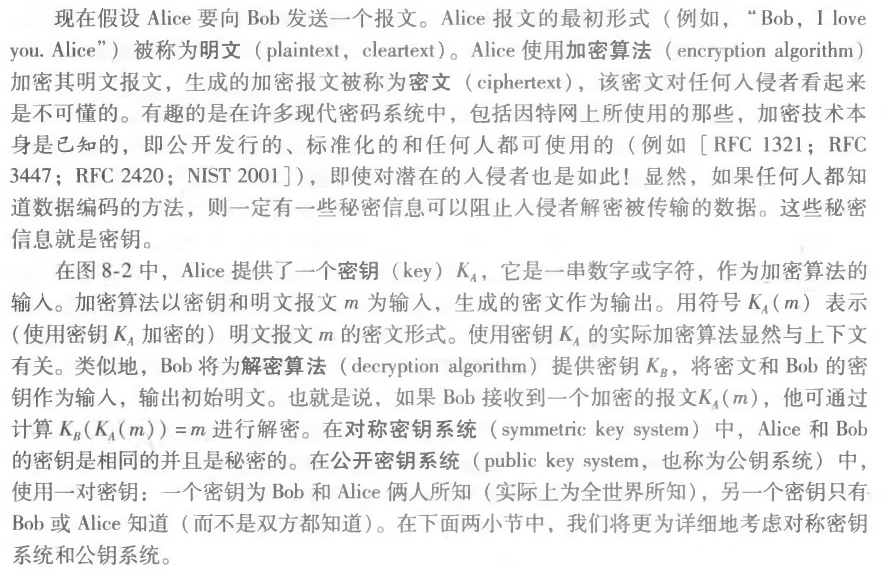
。。。。。。

## 参3：<https://en.wikipedia.org/wiki/Cryptography>

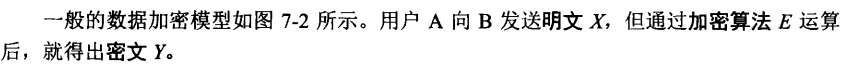
## 参4：《计算机网络-自顶向下方法》v7

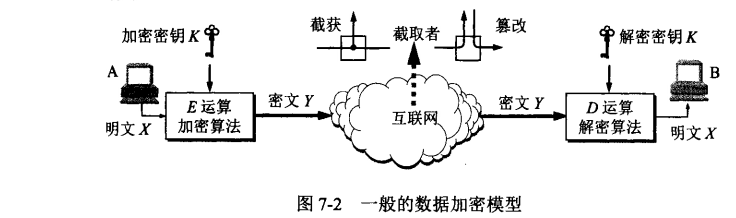


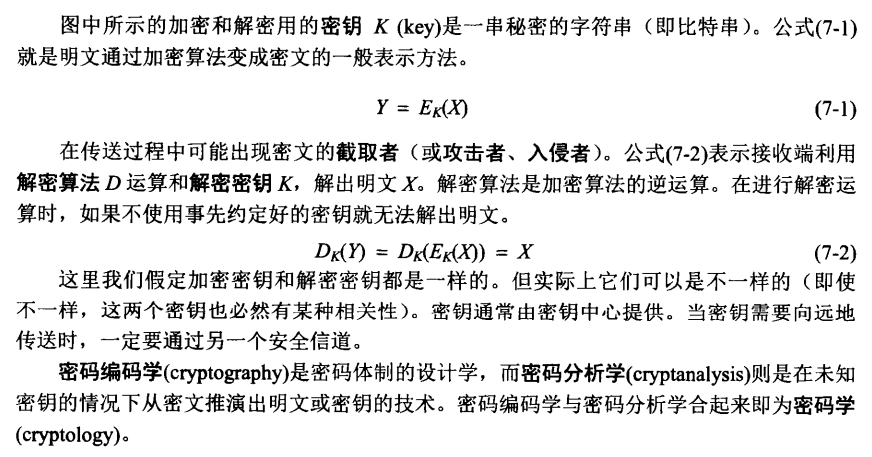


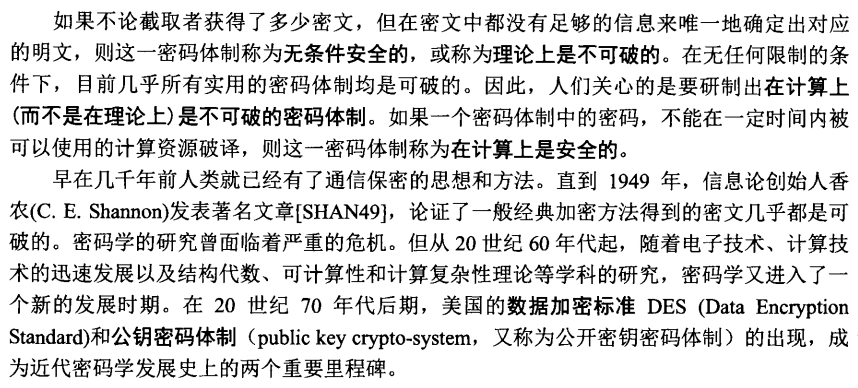


## 参5：《计算机网络 谢希仁》v7









## 归纳

### 【1】 常用术语的中英文

1. 明文：plaintext；
2. 密文：ciphertext；
3. 加密：encryption【名词】，encrypt/encipher【动词】；
4. 解密：decryption【名词】，decrypt/decipher【动词】；
5. 密码算法：cryptographic algorithm
6. 加密算法：encryption algorithm；
7. 解密算法：decryption algorithm；
8. 密钥：key；
9. 公钥：public key；
10. 私钥：private/secret key；

### 【2】 释义

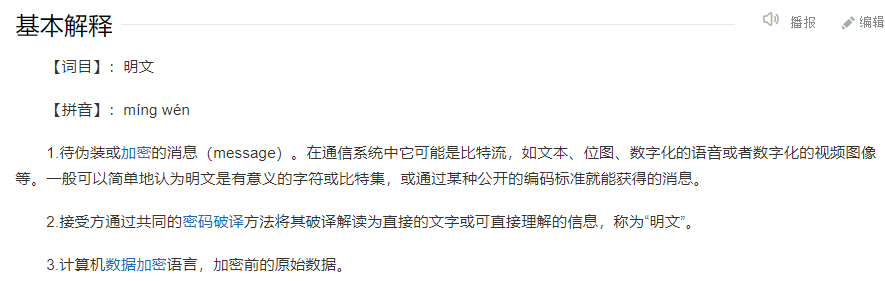
#### 《1》 明文

##### 参1：

<https://baike.baidu.com/item/%E6%98%8E%E6%96%87/9684633>

明文，是指没有加密的文字（或者[字符串](https://baike.baidu.com/item/%E5%AD%97%E7%AC%A6%E4%B8%B2/1017763)），一般人都能看懂的意思，属于[密码学](https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001)术语。[1]  在通信系统中它可能是[比特](https://baike.baidu.com/item/%E6%AF%94%E7%89%B9/3431582)流，如[文本](https://baike.baidu.com/item/%E6%96%87%E6%9C%AC/5443630)、[位图](https://baike.baidu.com/item/%E4%BD%8D%E5%9B%BE/1017781)、数字化的[语音](https://baike.baidu.com/item/%E8%AF%AD%E9%9F%B3/6140117)或者数字化的视频图像等。一般可以简单地认为明文是有意义的[字符](https://baike.baidu.com/item/%E5%AD%97%E7%AC%A6/4768913)或比特集，或通过某种公开的[编码](https://baike.baidu.com/item/%E7%BC%96%E7%A0%81/80092)标准就能获得的消息。经过某个[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86)算法进行作用，将作用后的文字称为[密文](https://baike.baidu.com/item/%E5%AF%86%E6%96%87/9684333)。对[密文](https://baike.baidu.com/item/%E5%AF%86%E6%96%87/9684333)来说，若想得到明文，就应通过与[加密算法](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86%E7%AE%97%E6%B3%95/2816213)对应的解密算法进行解密，恢复出明文。





##### 参2：<https://zh.wikipedia.org/wiki/%E6%98%8E%E6%96%87>

**明文**（Plaintext），在[密码学](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6)中是指传送方想要接收方获得的可读信息。 明文经过[加密](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)所产生的信息被称为[密文](https://zh.wikipedia.org/wiki/%E5%AF%86%E6%96%87)，而密文经过[解密](https://zh.wikipedia.org/wiki/%E8%A7%A3%E5%AF%86)还原得来的信息被称为明文。

##### 参3：

https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001

明文：没有进行加密，能够直接代表原文含义的信息。

#### 《2》 密文

##### 参1：

<https://baike.baidu.com/item/%E5%AF%86%E6%96%87/9684333>

密文是加了密的的文字，明文是加密之前的文字。密文是对明文进行加密后的报文[1]  。



##### 参2：<https://zh.wikipedia.org/wiki/%E5%AF%86%E6%96%87>

在[密码学](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6)中，**密文**（英语：ciphertext或cyphertext）是[明文](https://zh.wikipedia.org/wiki/%E6%98%8E%E6%96%87)经过[加密](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)[算法](https://zh.wikipedia.org/wiki/%E7%AE%97%E6%B3%95)所产生的。因为密文是一种除非使用恰当的算法进行[解密](https://zh.wikipedia.org/wiki/%E8%A7%A3%E5%AF%86)，人类或计算机不可以直接阅读理解的明文的形态，所以可以被理解为加密的信息。解密与加密是相对的，即一种使密文转化为明文的过程。

##### 参3：

https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001

密文：经过加密处理处理之后，隐藏原文含义的信息。

#### 《3》 加密

##### 参1：<https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86/752748>

加密，是以某种特殊的算法改变原有的[信息](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF/111163)数据，使得未授权的[用户](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7/3621489)即使获得了已加密的[信息](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF/111163)，但因不知[解密](https://baike.baidu.com/item/%E8%A7%A3%E5%AF%86/4409)的方法，仍然无法了解信息的内容。

在航空学中，指利用航空摄影像片上已知的少数控制点，通过对像片测量和计算的方法在像对或整条航摄带上增加控制点的[作业](https://baike.baidu.com/item/%E4%BD%9C%E4%B8%9A/5193971)。



##### 参2：<https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86>

在[密码学](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6)中，**加密**（英语：Encryption）是将[明文](https://zh.wikipedia.org/wiki/%E6%98%8E%E6%96%87)[信息](https://zh.wikipedia.org/wiki/%E4%BF%A1%E6%81%AF)改变为难以读取的[密文](https://zh.wikipedia.org/wiki/%E5%AF%86%E6%96%87)内容，使之不可读的过程。只有拥有解密方法的对象，经由解密过程，才能将密文还原为正常可读的内容。理想情况下，只有经授权的人员能够读取密文所要传达的信息。加密本身并不能防止信息传输被截取，但加密能防止截取者理解其内容。

因为种种技术原因，加密方法通常使用一个通过[算法](https://zh.wikipedia.org/wiki/%E7%AE%97%E6%B3%95)生成的[伪随机](https://zh.wikipedia.org/wiki/%E4%BC%AA%E9%9A%8F%E6%9C%BA)[密钥](https://zh.wikipedia.org/wiki/%E5%AF%86%E9%92%A5)。

虽然任何加密后的消息都可能被破解，但对于一个良好的加密算法而言，破解需要相当多的技术和算力。

授权读取信息的人可以轻松通过发信人所提供的密钥解密信息，但未经授权的人员则不行。

密码学历史中有众多加密方法；早期的加密方法常用于军事通讯。从此开始，现代计算中也出现了众多加密技术，并且加密在现代计算中也变得越来越常见。现代的加密方式通常使用[公钥](https://zh.wikipedia.org/wiki/%E5%85%AC%E9%92%A5)或[对称密钥](https://zh.wikipedia.org/wiki/%E5%AF%B9%E7%A7%B0%E5%AF%86%E9%92%A5)。

现代加密技术依赖现代计算机在破解密钥上并不高效的事实来保证其安全性。

##### 参3：

<https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001>

加密：将明文转换成密文的实施过程。

#### 《4》 解密

##### 参1：<https://baike.baidu.com/item/%E8%A7%A3%E5%AF%86/4409>

将“密文”变为“[明文](https://baike.baidu.com/item/%E6%98%8E%E6%96%87/9684633)”的过程被称为解密。



##### 参2：

<https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001>

解密：将密文转换成明文的实施过程。

#### 《5》 密码算法

#### 参1：

<https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001>

密码算法：密码系统采用的加密方法和解密方法，随着基于数学密码技术的发展，加密方法一般称为加密算法，解密方法一般称为解密算法。

#### 《6》 加密算法

##### 参1：

<https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86%E7%AE%97%E6%B3%95/2816213#2>

数据加密的基本过程就是对原来为明文的[文件](https://baike.baidu.com/item/%E6%96%87%E4%BB%B6/6270998)或数据按某种[算法](https://baike.baidu.com/item/%E7%AE%97%E6%B3%95/209025)进行处理，使其成为不可读的一段代码为“密文”， 使其只能在输入相应的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5/101144)之后才能显示出原容，通过这样的途径来达到保护数据不被非法人窃取、阅读的目的。该过程的逆过程为[解密](https://baike.baidu.com/item/%E8%A7%A3%E5%AF%86/4409)，即将该[编码](https://baike.baidu.com/item/%E7%BC%96%E7%A0%81/80092)信息转化为其原来数据的过程。



##### 参2：<https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86>

加密算法就是加密的方法。

加密算法可以分为两类：[对称加密](https://zh.wikipedia.org/wiki/%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)和[非对称加密](https://zh.wikipedia.org/wiki/%E9%9D%9E%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)。

在[密码学](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6)中，加密是将明文信息隐匿起来，使之在缺少特殊信息时不可读。

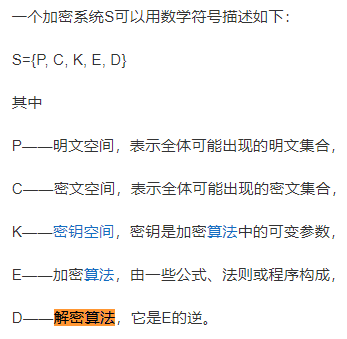
对称加密就是将信息使用一个密钥进行加密，解密时使用同样的密钥，同样的算法进行解密。

非对称加密，又称公开密钥加密，是加密和解密使用不同密钥的算法，广泛用于信息传输中。

#### 《7》 解密算法

##### 参1：

<https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86%E7%AE%97%E6%B3%95/2816213#2>



#### 《8》 密钥，公钥，私钥

##### 参1：<https://zh.wikipedia.org/wiki/%E5%AF%86%E9%92%A5>

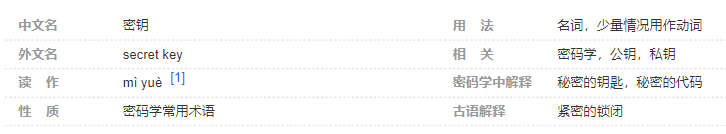
在[密码学](https://zh.wikipedia.org/wiki/%E5%AF%86%E7%A0%81%E5%AD%A6)中，**密钥**（key，又常称**金钥**）是指某个用来完成[加密](https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86)、[解密](https://zh.wikipedia.org/wiki/%E8%A7%A3%E5%AF%86)、完整性验证等密码学应用的秘密[信息](https://zh.wikipedia.org/wiki/%E4%BF%A1%E6%81%AF)。

在[对称密码学](https://zh.wikipedia.org/wiki/%E5%AF%B9%E7%A7%B0%E5%AF%86%E7%A0%81%E5%AD%A6)（或称密钥密码学）中，加密和解密用的是同一个钥匙，因此钥匙需要[保密](https://zh.wikipedia.org/wiki/%E4%BF%9D%E5%AF%86)。而在[公钥密码学](https://zh.wikipedia.org/wiki/%E5%85%AC%E9%92%A5%E5%AF%86%E7%A0%81%E5%AD%A6)（或称非对称密码学）中，加密和解密用的钥匙不同：通常一个是公开的，称为**公钥**；另一个保密，称为**私钥**。

现实世界中的钥匙通常是具有一定形状的金属片，钥匙用来打开锁。而密码算法中的“钥匙”（密钥）则用来“打开”密文。密钥是一串非常大的数字。根据密钥的使用方法，可将密码分为对称密码（symmetric cryptography）和非对称密码（asymmetric cryptography）两种。非对称密码通常又称为公钥密码（pubilc-key cryptography）。

##### 参2：<https://baike.baidu.com/item/%E5%AF%86%E9%92%A5/101144>

密钥是一种参数，它是在明文转换为密文或将密文转换为明文的算法中输入的参数。密钥分为对称密钥与非对称密钥。



##### 参3：

<https://baike.baidu.com/item/%E5%AF%86%E7%A0%81%E5%AD%A6/480001>

密钥：分为加密密钥和解密密钥。

# 两类密码体制

## 概述

### 参1：<https://zh.wikipedia.org/wiki/%E5%8A%A0%E5%AF%86>

现代的加密方式通常使用[公钥](https://zh.wikipedia.org/wiki/%E5%85%AC%E9%92%A5)或[对称密钥](https://zh.wikipedia.org/wiki/%E5%AF%B9%E7%A7%B0%E5%AF%86%E9%92%A5)。

### 参2：

<https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86/752748>

加密建立在对信息进行数学编码和解码的基础上。加密类型分为两种，[对称加密](https://baike.baidu.com/item/%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)与[非对称加密](https://baike.baidu.com/item/%E9%9D%9E%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)。

加密算法可以分为两类：[对称加密](https://baike.baidu.com/item/%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)和[非对称加密](https://baike.baidu.com/item/%E9%9D%9E%E5%AF%B9%E7%A7%B0%E5%8A%A0%E5%AF%86)。

## （2） 两类密码体制详解

参考：

1】 《计算机网络-第7版》谢希仁

### 【1】 对称密钥密码体制/对称加密

#### 《1》 是什么

参考：

1. 主：<https://www.kancloud.cn/cugfei/hadyang-interview/1317003>
2. 辅：谢希仁计网+维基百科+百度百科

**对称加密算法**的加密和解密使用的密匙：key是相同的，也就是说如果通讯两方使用对称加密算法来加密通讯数据，那么通讯双方就需要都知道这个密匙，收到通讯数据后用这个密匙来解密数据。

注：对称加密用的单个密钥也称为**对称密钥**

<https://learn.microsoft.com/zh-cn/windows/win32/seccrypto/symmetric-keys>

#### 《2》 过程



图示, 示意图

描述已自动生成

#### 《3》 子类

<https://www.cnblogs.com/ttss/p/4279757.html>

##### 1） DES

###### 1》 DES

###### 2》 三重DES

##### 2） AES

##### 3） RC4

<https://baike.baidu.com/item/RC4/3454548>

##### 相互对比

###### DES vs AES

<https://blog.csdn.net/m0_69916115/article/details/124668983>

1. AES与DES之间的主要区别在于加密过程。在DES中，将明文分为两半，然后再进行进一步处理；而在AES中，整个块不进行除法，整个块一起处理以生成密文。相对而言，AES比DES快得多，与DES相比，AES能够在几秒钟内加密大型文件。
2. 由于DES中使用的共享密钥的比特大小较小，因此它被认为不如AES安全。DES被认为更容易受到暴力攻击，而到目前为止，尚未遇到任何受严重攻击的AES。
3. 在灵活性的基础上评估算法的实现，AES比DES更具灵活性，因为它允许包括128、192、256位在内的各种长度的文本，而DES允许对64位固定文本进行加密。
4. DES回合处理中使用的功能是扩展，置换和替换，具有回合键的XOR操作，而AES回合中使用的功能是子字节，移位行，混合列和添加回合键。
5. AES实际上在硬件和软件实现上都是高效的，而DES最初只在硬件上有效。

### 【2】 公钥密码体制/非对称加密

#### 《1》 是什么

参考：

1. 主：<https://www.kancloud.cn/cugfei/hadyang-interview/1317003>
2. 辅：谢希仁计网+维基百科+百度百科

它需要两个密钥，**一个是公开密钥，另一个是私有密钥；一个用作加密的时候，另一个则用作解密**。使用其中一个密钥把明文加密后所得的密文，只能用相对应的另一个密钥才能解密得到原本的明文。由于加密和解密需要两个不同的密钥，故被称为非对称加密。

虽然两个密钥在数学上相关，但如果知道了其中一个，并不能凭此计算出另外一个。因此其中一个可以公开，称为 **公钥**，任意向外发布；不公开的密钥为 **私钥** ，必须由用户自行严格秘密保管，绝不透过任何途径向任何人提供，也不会透露给要通信的另一方，即使他被信任。

公钥 & 私钥 均可以作为加密密钥。

注：非对称加密用的公钥/私钥对也称为**非对称密钥**

<https://learn.microsoft.com/zh-cn/windows/win32/seccrypto/public-private-key-pairs>

#### 《2》 为什么

##### 参1：《计算机网络-v7谢希仁》

手机屏幕截图

描述已自动生成

#### 《3》 过程

##### 参1：《计算机网络-v7谢希仁》

文本, 信件

描述已自动生成

图示

描述已自动生成

#### 《4》 子类

##### RSA

##### DSA

##### 椭圆曲线

##### 相互对比

###### RSA vs DSA

参考：

1. 以下两篇内容相同

<https://zhuanlan.zhihu.com/p/584307289>

<https://www.freebuf.com/news/350065.html>

1. <https://cloud.tencent.com/developer/news/254061>

虽然 RSA 和 DSA 使用不同类型的数学算法来生成密钥对，但出于加密强度的目的，两者被认为是等效的。相反，RSA 和 DSA 之间的主要区别在于性能和速度。

性能和速度

RSA 在加密和签名方面比 DSA 快，但在解密和验证方面比 DSA 慢。但是，由于身份验证需要两者，因此对于许多实际应用程序而言，性能差异在很大程度上可以忽略不计。

RSA 在密钥生成方面也比 DSA 慢，但由于密钥生成一次并使用数月或数年，这通常不是一个重要的考虑因素。

SSH 协议支持

另一个不同之处在于它们的 Secure Shell (SSH) 协议支持。 RSA 与原始 SSH 以及更新的第二版 SSH2 兼容，而 DSA 仅适用于 SSH2。由于 SSH 被认为不如 SSH2 安全，这可能是使用 DSA 的一个考虑因素。

联邦背书

DSA 和 RSA 之间的另一个区别是 DSA 得到美国联邦政府的认可。 对于向联邦机构提供服务的企业，跟上政府标准的能力可能是使用 DSA 的一个理由。

底线是，对于大多数用例、行业和监管环境，RSA 和 DSA 非常相似，提供相同的加密强度，两者之间的差异相对较小。 这两种算法也同样兼容领先的互联网协议，包括 Nettle、OpenSSL、wolfCrypt、Crypto++ 和 cryptlib。

#### 《5》 PKI《Public Key Infrastructure》：公钥基础设施

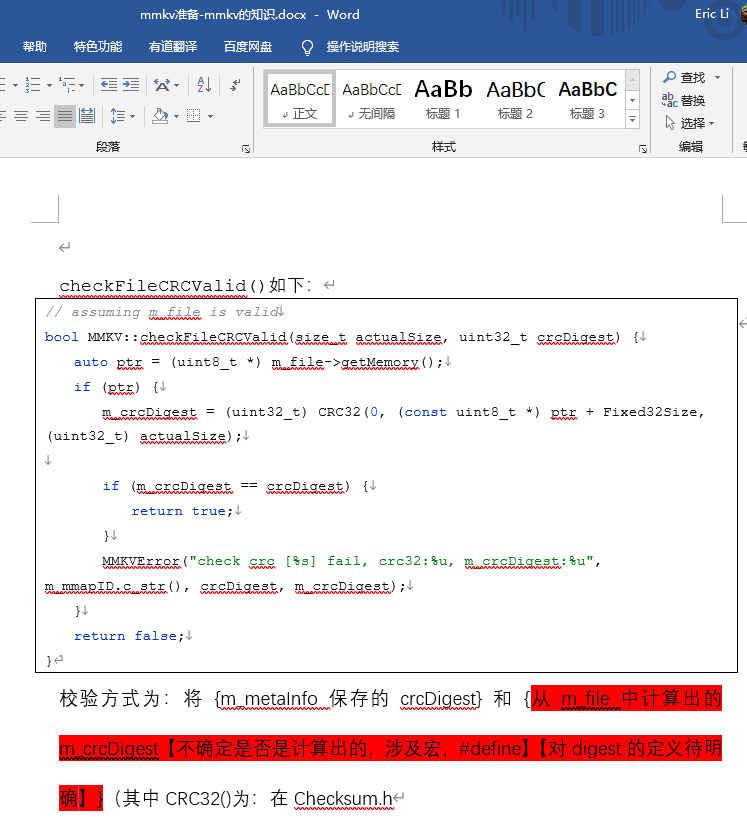
<https://baike.baidu.com/item/%E5%85%AC%E9%92%A5%E5%9F%BA%E7%A1%80%E8%AE%BE%E6%96%BD/10881894>

# 数字签名

和消息摘要的关系

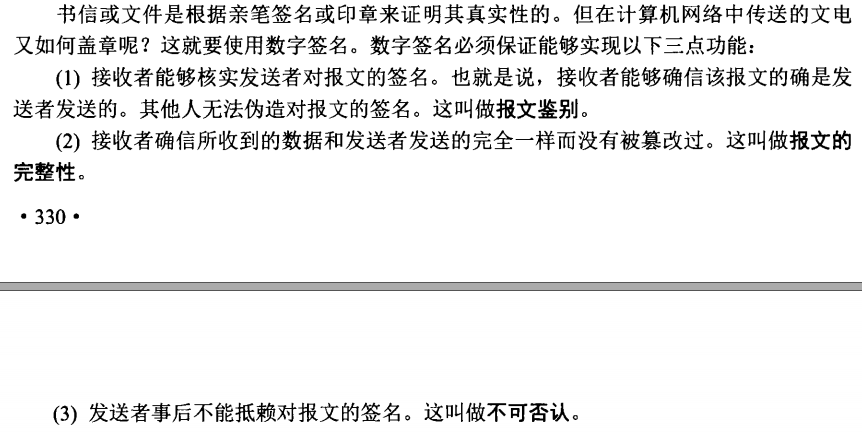
https://www.jianshu.com/p/230995f176b1

示例：



## 是什么

### 参1：《计算机网络 谢希仁》v7

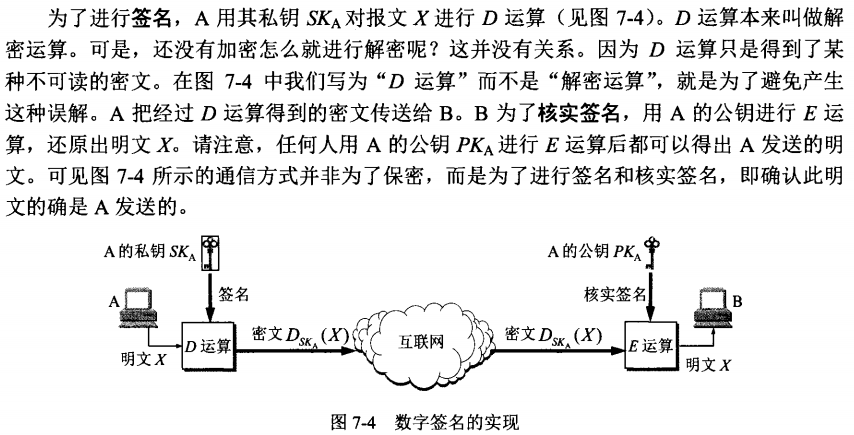


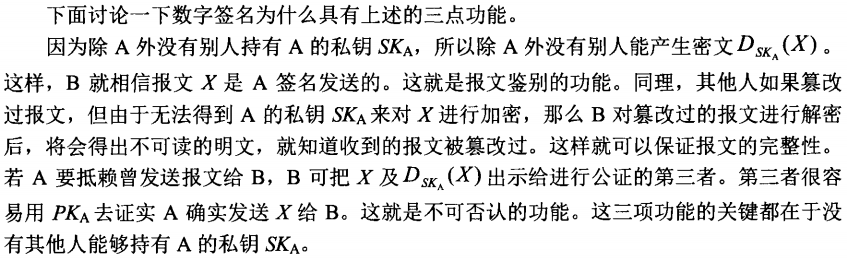
## 过程

### 参1【无消息摘要】：《计算机网络 谢希仁》v7

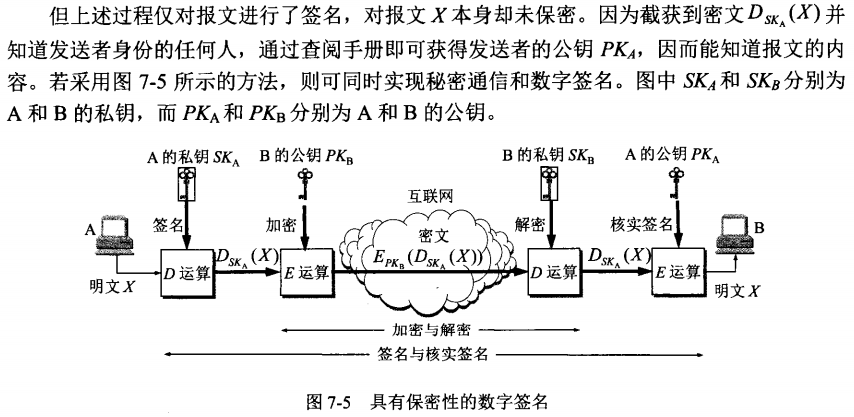


#### 普通





#### 加强



### 参2【有消息摘要】： 见“本文档”-消息摘要（2）

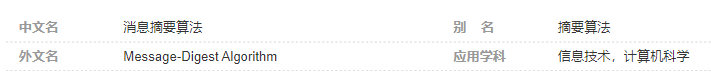
# 消息摘要：Message Digest

## 概述

### 参1：

<https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81%E7%AE%97%E6%B3%95/3286770>

消息摘要算法的主要特征是[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86/752748)过程不需要[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5/101144)，并且经过加密的数据无法被解密，可以被解密逆向的只有[CRC32](https://baike.baidu.com/item/CRC32/7460858)算法，只有输入相同的明文数据经过相同的消息摘要算法才能得到相同的[密文](https://baike.baidu.com/item/%E5%AF%86%E6%96%87/9684333)。



### 参2：<https://en.wikipedia.org/wiki/Digest>



### 参3：

<https://en.wikipedia.org/wiki/Cryptographic_hash_function>

#### 定义：

A **cryptographic**[**hash function**](https://en.wikipedia.org/wiki/Hash_function) (**CHF**) ：加密哈希函数/加密散列函数is a mathematical [algorithm](https://en.wikipedia.org/wiki/Algorithm) that [maps](https://en.wikipedia.org/wiki/Map_(mathematics)) data of an arbitrary size (often called the "message") to a [bit array](https://en.wikipedia.org/wiki/Bit_array) of a fixed size (the "hash value", "hash", or "message digest").

#### 不可逆性：

It is a [one-way function](https://en.wikipedia.org/wiki/One-way_function), that is, a function for which it is practically infeasible to invert or reverse the computation. Ideally, the only way to find a message that produces a given hash is to attempt a [brute-force search](https://en.wikipedia.org/wiki/Brute-force_search) of possible inputs to see if they produce a match, or use a [rainbow table](https://en.wikipedia.org/wiki/Rainbow_table) of matched hashes.

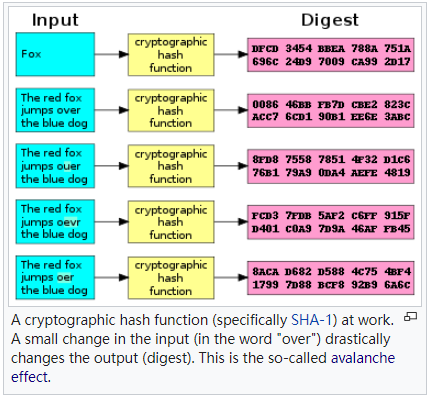
#### 和现代加密的关系：

Cryptographic hash functions are a basic tool of modern cryptography.

#### 特性：

A cryptographic hash function must be [deterministic](https://en.wikipedia.org/wiki/Deterministic_algorithm), meaning that the same message always results in the same hash. Ideally it should also have the following properties:

* it is quick to compute the hash value for any given message
* it is infeasible to generate a message that yields a given hash value (i.e. to reverse the process that generated the given hash value)
* it is infeasible to find two different messages with the same hash value
* a small change to a message should change the hash value so extensively that a new hash value appears uncorrelated with the old hash value ([avalanche effect](https://en.wikipedia.org/wiki/Avalanche_effect))



#### 应用：

Cryptographic hash functions have many [information-security](https://en.wikipedia.org/wiki/Information_security) applications, notably in [digital signatures](https://en.wikipedia.org/wiki/Digital_signature), [message authentication codes](https://en.wikipedia.org/wiki/Message_authentication_code) (MACs), and other forms of [authentication](https://en.wikipedia.org/wiki/Authentication). They can also be used as ordinary [hash functions](https://en.wikipedia.org/wiki/Hash_function), to index data in [hash tables](https://en.wikipedia.org/wiki/Hash_table), for [fingerprinting](https://en.wikipedia.org/wiki/Fingerprint_(computing)), to detect duplicate data or uniquely identify files, and as [checksums](https://en.wikipedia.org/wiki/Checksum) to detect accidental data corruption. Indeed, in information-security contexts, cryptographic hash values are sometimes called (*digital*) *fingerprints*, *checksums*, or just *hash values*, even though all these terms stand for more general functions with rather different properties and purposes.

## 和数字签名的关系

### 参1：

<https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81%E7%AE%97%E6%B3%95/3286770>

消息摘要算法主要应用在“数字签名”领域，作为对明文的摘要算法。

### 参2：

<https://en.wikipedia.org/wiki/Cryptographic_hash_function>

**Signature generation and verification[**[**edit**](https://en.wikipedia.org/w/index.php?title=Cryptographic_hash_function&action=edit&section=6)**]**

*Main article:*[*Digital signature*](https://en.wikipedia.org/wiki/Digital_signature)

Almost all [digital signature](https://en.wikipedia.org/wiki/Digital_signature) schemes require a cryptographic hash to be calculated over the message. This allows the signature calculation to be performed on the relatively small, statically sized hash digest. The message is considered authentic if the signature verification succeeds given the signature and recalculated hash digest over the message. So the message integrity property of the cryptographic hash is used to create secure and efficient digital signature schemes.：几乎所有的数字签名方案都需要在消息上计算一个加密哈希。这允许在相对较小的、静态大小的哈希摘要上执行签名计算。如果签名验证成功给出了该签名，并在消息上重新计算哈希摘要，则认为该消息是可信的。因此，加密哈希的消息完整性属性被用来创建安全和高效的数字签名方案。

### 参3：

<https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81%E7%AE%97%E6%B3%95/3286770>

#### 常见情况：

几乎所有的[数字签名](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E7%AD%BE%E5%90%8D)方案都要和快速高效的摘要算法（Hash函数）一起使用，当[公钥算法](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5%E7%AE%97%E6%B3%95)与摘要算法结合起来使用时，便构成了一种有效地数字签名方案。

#### 过程：

这个过程是：首先用摘要算法对消息进行摘要，然后在把摘要值用信源的[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5)加密；接收方先把接收的明文用同样的摘要算法摘要，形成“准签体”， 然后再把准签体与用信源的[公钥](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5)解密出的“签体”进行比较，如果相同就认为消息是完整的，否则消息不完整。

#### 好处：

这种方法使公钥加密只对消息摘要进行操作，因为一种摘要算法的摘要消息长度是固定的，而且都比较“短”（相对于消息而言），正好符合公钥加密的要求。这样效率得到了提高，而其安全性也并未因为使用摘要算法而减弱。

## 常见算法

<https://en.wikipedia.org/wiki/Cryptographic_hash_function#Cryptographic_hash_algorithms>

There are many cryptographic hash algorithms; this section lists a few algorithms that are referenced relatively often. A more extensive list can be found on the page containing a [comparison of cryptographic hash functions](https://en.wikipedia.org/wiki/Comparison_of_cryptographic_hash_functions).

### 【1】 MD5

### 【2】 SHA-1

### 等等

# 哈希/散列（函数）：Hash（Function）

## （1） 是什么

<https://baike.baidu.com/item/Hash/390310>

<https://zh.wikipedia.org/wiki/%E6%95%A3%E5%88%97%E5%87%BD%E6%95%B8>

**散列函数**，又称**散列算法**、**哈希函数**，是一种把任意长度的输入变换成固定长度的输出的方法，该输出就是**散列值/哈希值**：hash value。

## （2） 性质

<https://zh.wikipedia.org/wiki/%E6%95%A3%E5%88%97%E5%87%BD%E6%95%B8#%E6%95%A3%E5%88%97%E5%87%BD%E6%95%B0%E7%9A%84%E6%80%A7%E8%B4%A8>

1. 所有散列函数都有如下一个基本特性：如果两个散列值是不相同的（根据同一函数），那么这两个散列值的原始输入也是不相同的。这个特性是散列函数具有[确定性](https://zh.wikipedia.org/wiki/%E7%A1%AE%E5%AE%9A%E6%80%A7)的结果，具有这种性质的散列函数称为单向散列函数。

但另一方面，散列函数的输入和输出不是唯一对应关系的，如果两个散列值相同，两个输入值很可能是相同的，但也可能不同，这种情况称为“[**散列碰撞**](https://zh.wikipedia.org/w/index.php?title=%E9%9B%9C%E6%B9%8A%E7%A2%B0%E6%92%9E&action=edit&redlink=1)**（collision）**”，这通常是两个不同长度的输入值，刻意计算出相同的输出值。

1. 在密码学中，散列函数必须具有不可逆性。

## （3） 常见散列算法/函数

<https://zh.wikipedia.org/wiki/%E6%95%A3%E5%88%97%E5%87%BD%E6%95%B8#%E7%9B%AE%E5%89%8D%E5%B8%B8%E8%A6%8B%E7%9A%84%E9%9B%9C%E6%B9%8A%E6%BC%94%E7%AE%97%E6%B3%95>

### 【1】 MD5

### 【2】 SHA-1

## （x） 应用

### 【1】 散列/哈希表：Hash Table

#### 《1》 是什么

参考：

1. 《2021王道数据结构》
2. <https://baike.baidu.com/item/Hash/390310#3>

**散列表**，是根据关键字：key而直接进行访问的数据结构。也就是说，散列表通过**散列函数**建立了关键字和存储地址之间的一种直接映射关系。

这里的散列函数可表示为：Hash（key）=Addr。Addr表示的地址可以是数组下标、索引或内存地址等。

理想情况下，对散列表进行查找的时间复杂度为O（1），即与表中元素的个数无关。

#### 《2》 散列表中常用的散列函数

#### 《3》 散列表中的哈希碰撞/冲突

参考：

1】 《2021王道数据结构》

2】 <https://baike.baidu.com/item/Hash/390310#2>

3】 <https://blog.csdn.net/zsyoung/article/details/114505480>

4】 <https://cloud.tencent.com/developer/article/1776352>

##### 1） 是什么

散列函数可能会把两个或两个以上的不同关键字（key）映射到同一地址，称这种情况为**哈希碰撞/冲突**。这些发生碰撞的不同关键字称为**同义词**。

一方面，设计得好的散列函数应尽量减少这样的冲突；另一方面，由于这样的冲突总是不可避免的，所以还要设计好处理冲突的方法，即为产生冲突的关键字寻找下一个“空”的Hash地址。

##### 2） 如何解决

常用的Hash冲突解决方法有以下几种：

###### 1》 开放寻址法/开放定址法

这种方法的基本思想是：当关键字key的哈希地址p=H（key）出现冲突时，以p为基础，产生另一个哈希地址p1，如果p1仍然冲突，再以p1为基础，产生另一个哈希地址p2，…，直到找出一个不冲突的哈希地址pi，将相应元素存入其中。

其数学递推公式为

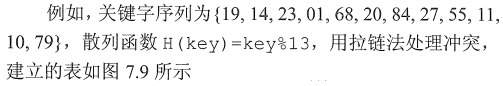


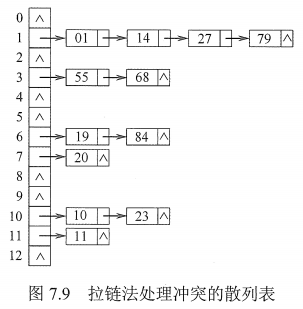
取定某一增量序列后，对应的处理方法就是确定的。增量序列通常有以下取法：

* 线性探测法
* 平方探测法
* 再散列法
* 伪随机序列法

###### 2》 拉链法/链地址法

这种方法的基本思想是将所有哈希地址为i的元素构成一个称为同义词链的单链表，并将单链表的头指针存在哈希表的第i个单元中，因而查找、插入和删除主要在同义词链中进行。链地址法适用于经常进行插入和删除的情况。





#### 《4》 装填/加载/负载因子

见

#### 《5》 性能分析

参考：《2021王道数据结构》



# 一些注意点

## （1） 加密和编码

### 参1：<https://www.codenong.com/4657416/>

#### 回答1：

* 作为第一种近似，加密使用密钥，而编码则不使用。

#### 回答2：

编码：

 目的：编码的目的是转换数据，以便不同类型的系统能够正确(安全)地使用数据。

 用于：维护数据可用性，即确保数据能够被正确使用。

 数据检索机制：如果我们知道编码中使用的是什么算法，就没有密钥，很容易被颠倒。

 使用的算法：ascii、unicode、url编码、base64。

 示例：通过电子邮件发送的二进制数据，或在网页上查看特殊字符。

加密：

 目的：加密的目的是为了对数据进行转换，使其不被他人知道。

 用于：维护数据机密性，即确保数据不会被预期收件人以外的任何人使用。

 数据检索机制：如果我们知道所使用的密钥和加密算法，就可以获得原始数据。

 使用的算法：AES、Blowfish、RSA。

 例如：向某人发送一封只有他们才能阅读的秘密信，或者通过互联网安全地发送密码。

#### 回答3：

我想说，两种操作都将信息从一种形式转换为另一种形式，区别在于：

* 编码意味着将信息从一种形式转换为另一种形式，在大多数情况下，它很容易可逆。
* 加密意味着原始信息被屏蔽，并涉及加密密钥，必须向加密/解密过程提供加密密钥才能进行转换。

所以，如果它涉及(对称或非对称)密钥(又称"秘密")，那么它就是加密，否则它就是编码。