《现代密码学》实验指导书

实验1：AES工作方式实现

**一、实验目的**

掌握AES密码系统的加密解密工作模式，掌握AES在OFB和CBC下的工作原理和差异。

**二、AES的工作模式**

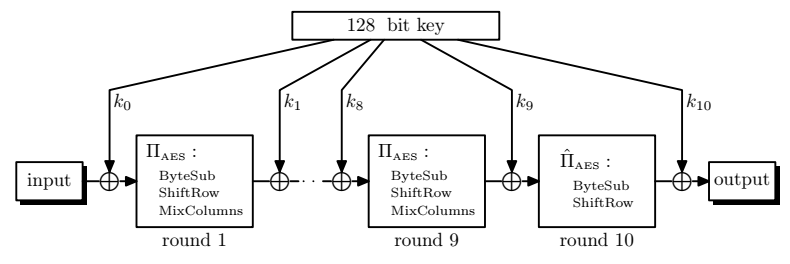
1. AES的工作原理

AES算法是一个迭代型分组密码，采用的是代替置换网络，即SP结构。其明文分组长度分别有128，192和256位，密钥长度可取28，192和256 位。相应的加密轮数分别为10轮，12轮，14轮，加密过程中分别需要11个，13个，15个子密钥。我们将介绍明文及密钥长度都是128 位的Rijndael 密码体制，加密轮数为10 ，子密钥为11 个，记，，…， 这也是目前使用最广泛的方式。

AES加密过程的第1轮到第9轮的轮函数一样，包括4个操作：字节代换，行移位，列混合，加上密钥。最后一轮迭代不执行列混合。另外在第1轮迭代之前，先将明文用原始密钥进行一次加上密钥操作。

由于没有使用Feistel 结构， AES 的解密过程和加密过程不同，需要使用相应变换的逆变换，为此首先需要定义加密算法中每轮中四种基本操作的逆操作。AES中的解密方案不同于DES和IDEA中的解密方案。

AES解密方案中每轮都使用4个变换：逆行移位，逆字节代换，加上密钥，逆列混合。这要求AES的加密解密算法需要使用两个不同的软件或硬件模块。另外还有一种等价的解密方案，它使得解密过程中各个变换的使用顺序是一致的，只是将计算部件转换为逆部件。这种方案利用了逆行移位和逆字节代换顺序可以交换，加上密钥和列混合都是针对状态矩阵的列进行操作。



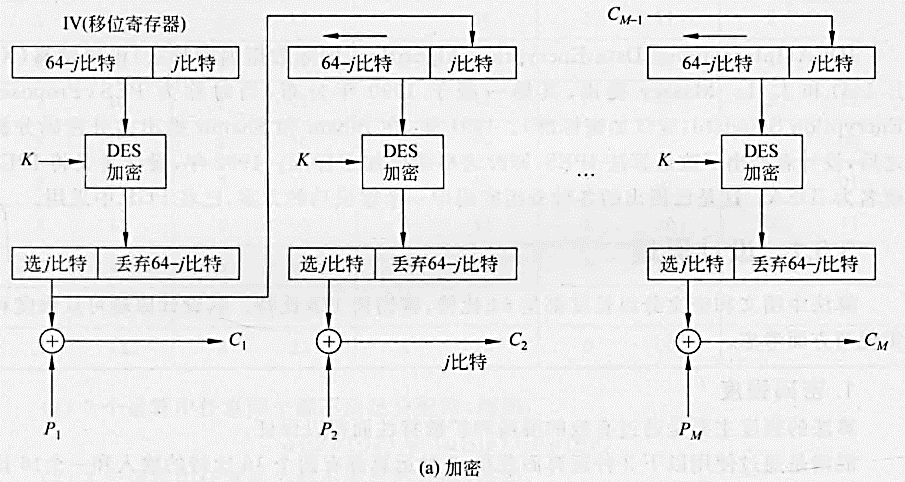
2. CBC工作模式

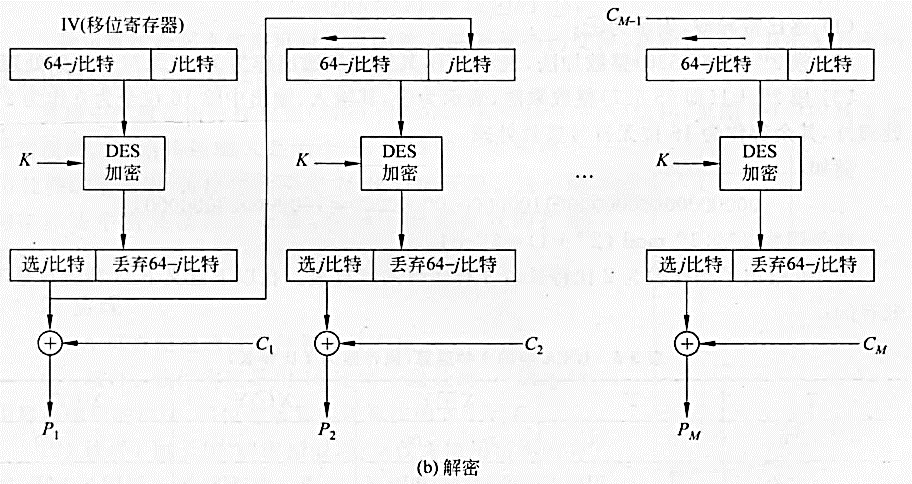
为了解决ECB的安全缺陷， CBC模式可以让重复的明文分组产生不同的密文分组。右图是CBC模式示意图，它一次对一个明文分组加密，每次加密使用同一密钥，加密算法的输入是当前明文分组和前一次密文分组的异或，因此加密算法的输入不会显示出与这次的明文分组之间的固定关系，所以重复的明文分组不会在密文中暴露出这种重复关系。



3. OFB工作模式

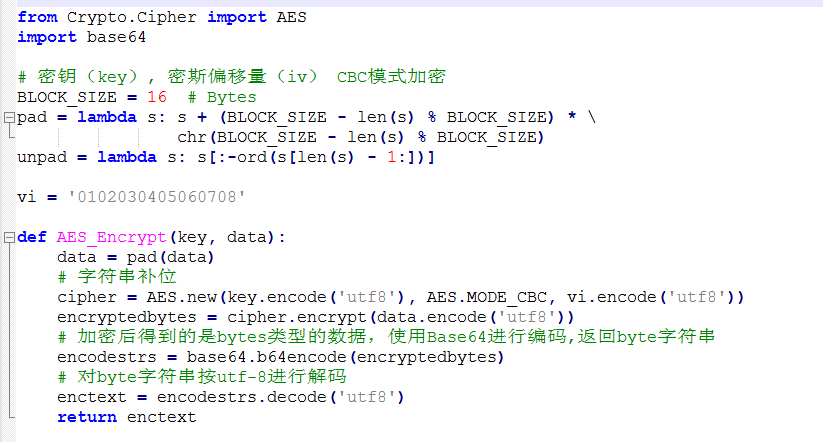
OFB（Output Feedback）模式的结构类似于CFB。不同之处如下：OFB模式是将加密算法的输出反馈到移位寄存器，而CFB模式中是将密文单元反馈到移位寄存器。OFB模式的优点是传输过程中的比特错误不会被传播。例如中出现一比特错误，在解密结果中只有受到影响，以后各明文单元则不受影响。而在CFB中，也作为移位寄存器的输入，因此它的一比特错误会影响解密结果中各明文单元的值。

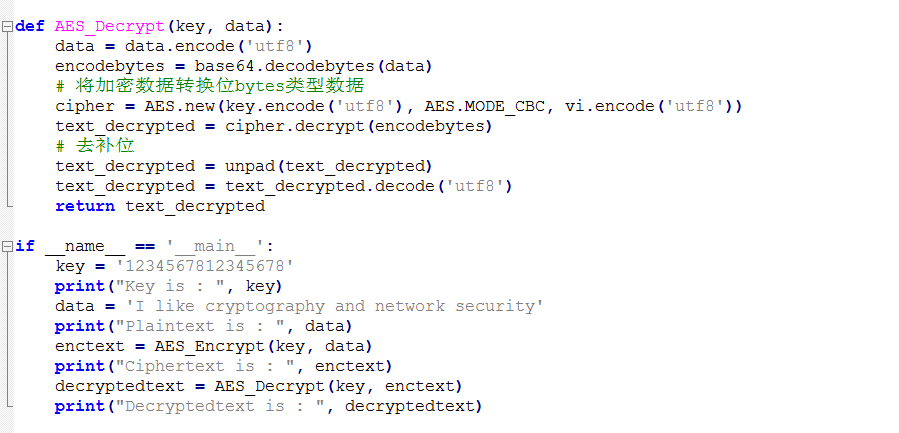




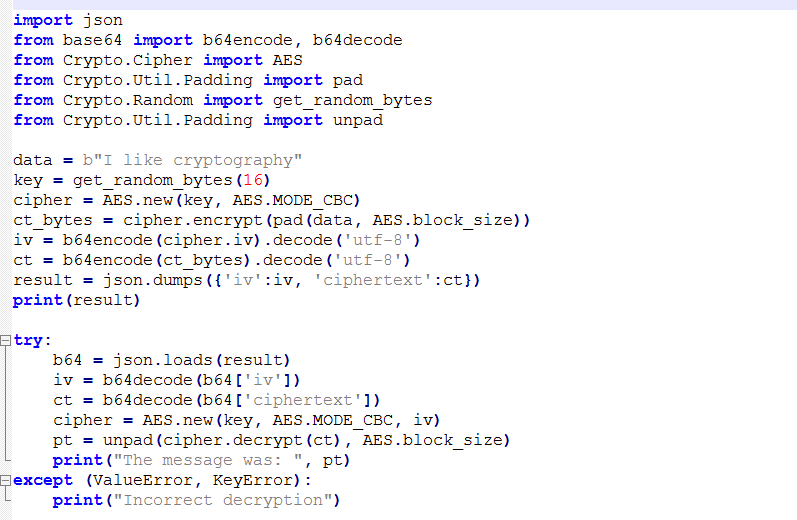
三、实验程序

1. AES加密和解密

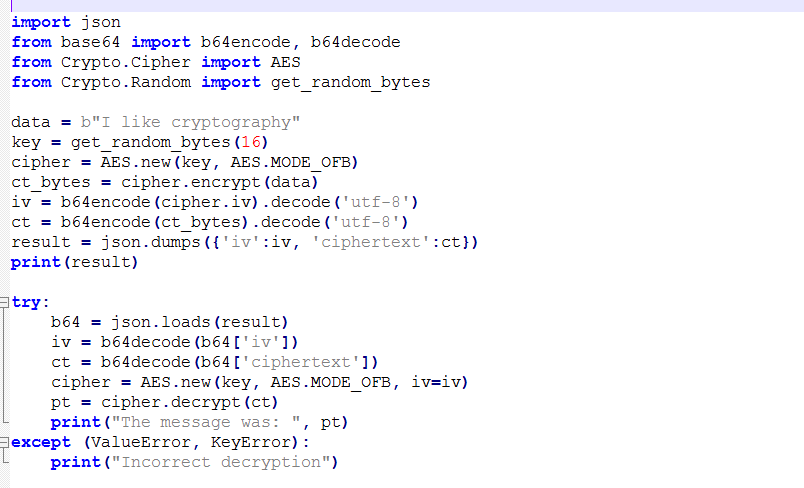




2. AES-CBC模式



3. AES-OFB模式



实验2：简单RSA加密算法实现

一、实验目的

掌握RSA加密和解密的实现过程。掌握RSA密钥产生方法。

二、RSA的工作原理

1. RSA的密钥产生

RSA的密钥产生过程如下：

（1）选两个保密的大素数和；

（2）计算和欧拉函数；

（3）选一整数，满足，且；

（4）计算，满足；由于和互素，因此在模下的乘法逆元一定存在。

（5）以为公开钥，为秘密钥。

2. 加密

加密时首先将明文比特串分组，使得每个分组对应的十进制数小于，即分组长度小于。然后对每个明文分组，作加密运算：

3. 解密

对密文分组的解密运算为：

三、实验程序

1. RSA-1024加密和解密程序



1. RSA密钥生成





实验3：MD5和SHA1哈希函数实验

一、实验目的

掌握MD5和SHA1哈希函数的工作原理，掌握不同参数对哈希值的影响，掌握不同方法实现哈希函数计算的方法。比较MD5和SHA1的差异。

二、MD5和SHA1的工作原理

MD5摘要算法包括以下过程。

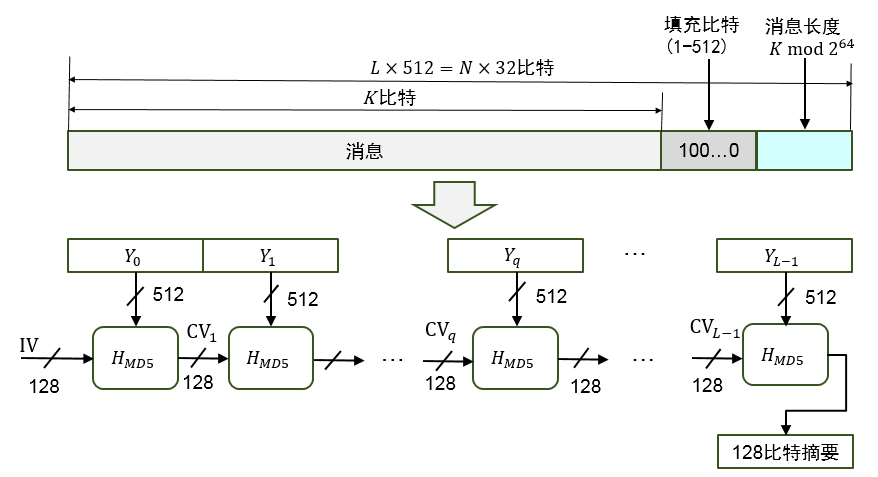
（1）对消息进行填充：使得消息比特长度在模512为448，即长度为，预留的64比特是供（2）不使用。

（2）附加消息长度：利用第（1）步预留的64比特以小端方式（低有效字节存放低地址字节）表示被填充前消息的长度，如果长度超过，则以取模。

（3）对128比特长MD5数据缓冲区初始化。

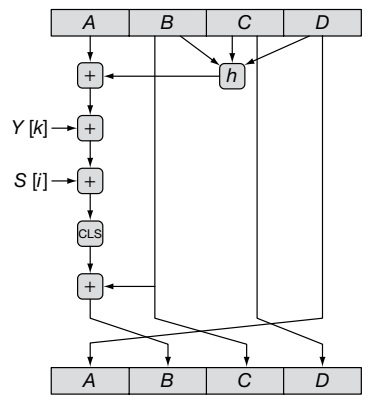
（4）以分组为单位对消息进行处理。

（5）输出：消息的所有分组处理结束后最后一个的输出就是消息摘要。



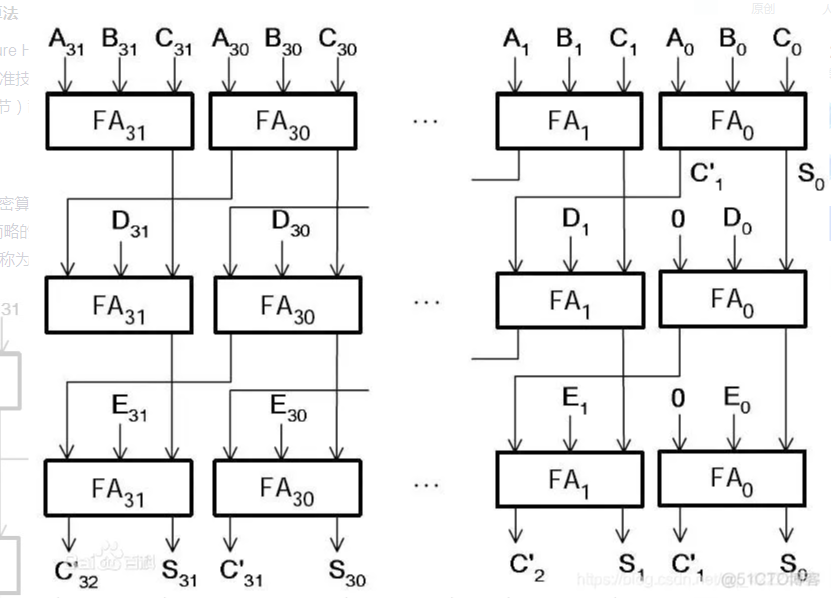
MD5数据压缩过程如图所示。

其中：表示32位循环左移位，表示表中第个字，为消息第个分组中第个字。



三、SHA1的工作原理

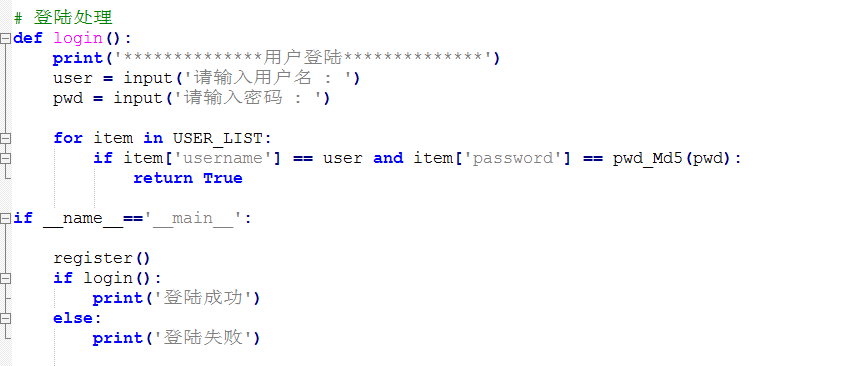
SHA-1是一种密码哈希函数，美国国家安全局设计，并由美国国家标准技术研究所（NIST）发布为联邦数据处理标准（FIPS）。SHA-1可以生成一个被称为消息摘要的160位（20字节）哈希值，哈希值通常的呈现形式为40个十六进制数。



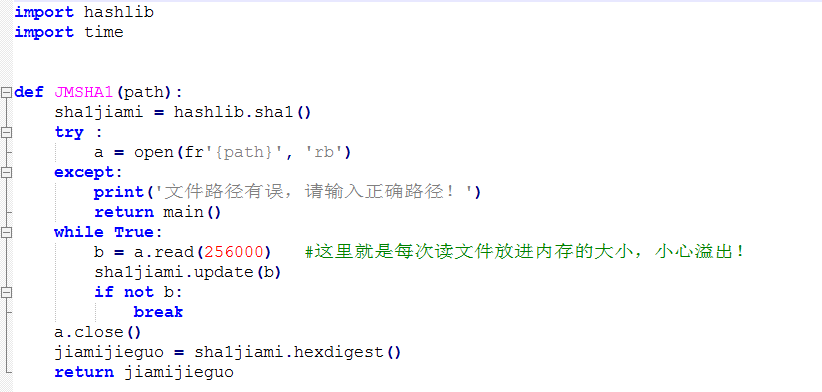
四、实验程序

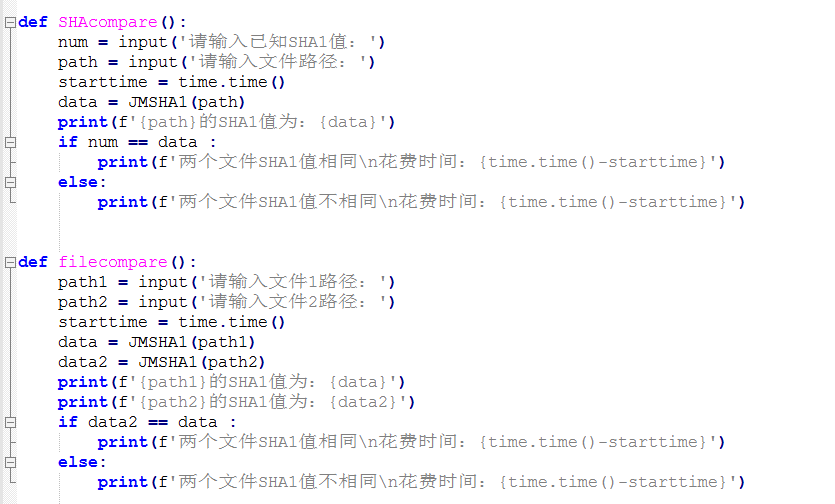
1. MD5实现门户登陆





2. SHA1实验







实验4：数字签名实验

1. 实验目的

掌握离散对数和椭圆曲线数字签名的实现过程。

1. DSA的原理
2. 基于离散对数的数字签名

DSA是在ElGamal和Schnorr两个签名方案的基础上设计的，其安全性基于求离散对数的困难性。算法描述如下。

（1）全局公开钥

：满足的大素数，其中，且是64的倍数。

：的素因子，满足，即长为160比特。

：，其中是满足，且满足的任一整数 。

（2）用户秘密钥

是满足的随机数或伪随机数。

（3） 用户的公开钥

（4）用户为待签消息选取的秘密数

是满足 的随机数或伪随机数。

（5）签名过程

利用SHA计算哈希值，然后计算

用户对消息的签名为。

1. 基于椭圆曲线的数字签名

ECDSA是由素域上的椭圆曲线进行数字签名。ECDSA的全局域参数如下：

（1）：一个素数；

（2），：上的整数，由方程定义椭圆曲线；

（3）：椭圆曲线上的基点，定义为；

（4）：点的阶，即满足的最小整数，等于曲线上有理点的个数。

数字签名需要产生一个密钥对，包括一个公钥和一个私钥。签名者Bob通过下面的方式产生两个密钥：

（1）选择随机整数，；

（2）计算，得到曲线上另一个有理点；

（3）Bob的公钥是，私钥是。

通过使用公开的全局域参数和私钥，BoB对消息产生320字节的数字签名，过程如下：

（1）选择随机或伪随机数，其中；

（2）计算曲线上的有理点和。如果，则转到步骤（1）；

（3）计算；

（4）计算，其中是哈希函数SHA-2或SHA-3；

（5）计算。如果，则转到步骤（1）；

（6）消息的签名是。

1. 实验程序
2. RSA数字签名



1. ECC数字签名

