文章编号:1004-1478(2011)06-0046-06

基于MFC的AES可视化试验平台的设计与实现

乔子芮[1], 周彦伟[2]

(1.陕西师范大学 计算机科学学院, 陕西 西安 710062;
2.陕西师范大学 教师专业能力发展中心, 陕西 西安 710062)

摘要:为了解高级加密标准AES的加/解密操作过程,基于MFC设计并实现了AES算法的可视化试验平台.该平台动态显示AES加密过程中每一阶段密文和密钥的变换情况,通过再现AES加/解密的工作方式,使复杂的迭代过程更容易理解,同时实现了对文件的加/解密操作.

关键词:Microsoft基本类库;高级加密标准;可视化试验平台;密钥
中图分类号:TP309 **文献标志码**:A

Design and implementation of AES's visualization platform based on MFC

QIAO Zi-rui[1], ZHOU Yan-wei[2]

(1. *School of Comp. Sci., Shaanxi Normal Univ., Xi'an* 710062, *China*;
2. *Center for Teacher Professional Ability Development, Shaanxi Normal Univ., Xi'an* 710062, *China*)

Abstract: To understand the encryption and decryption operation of AES, the visualization platform of AES was designed and implementation based on MFC. It shows the transformation of the ciphertext and the key in every stage of AES's encryption process, through reproduction AES's encryption/decryption process we can easily understand the complex iterative process, the document encryption/decryption operations are achieved.

Key words: Microsoft foundation class (MFC); advanced encryption standard (AES); visualization platform; key

0 引言

随着信息技术和互联网技术的发展与普及,网络通信已经渗透到社会的各个方面.信息安全和隐私的保护程度已成为衡量互联网关键技术先进性的指标.文献[1]中的民意调查表明,用户使用互联网时感到最大的问题是担心自己的隐私被发现.那么如何保证信息在发送方和接收方之间传送时不被窃密者窃取并破译,是我们所面对的一个重要问题.要想使信息可靠传输,发信者必须对所发的数据(即明文)进行加密,使其变成密文,收信者收到密文后再用相应的解密系统对密文解密,恢复出明

收稿日期:2011-03-25
基金项目:国家863基金资助项目(2007AA01Z438200)
作者简介:乔子芮(1985—),女,陕西省定边县人,陕西师范大学硕士研究生,主要研究方向为数据挖掘、粒计算.

文[2]. 加解密过程中密码的安全性对信息的安全级别起决定性作用.

文献[2]基于 MFC(Microsoft foundation class)开发出 DES(data encryption standard)分组密码算法的可视化平台，再现了 DES 的加解密过程. 随着对称密码的发展，DES 数据加密标准算法由于密钥长度较小(56 b)，已经不适应当今分布式开放网络对数据加密安全性的要求. 因此 1997 年 NIST(美国国家标准和技术协会)公开征集新的数据加密标准，即高级加密标准 AES(advanced encryption standard). 经过 3 轮的筛选，比利时的 Joan Daeman 等提交的 Rijndael 算法被提议为 AES 的最终算法，于 2002 年 5 月 26 日正式生效. 此算法成为美国新的数据加密标准而被广泛应用在各个领域. 尽管人们对 AES 还有不同的看法，但总体说，作为新一代的数据加密标准，AES 汇聚了强安全性、高性能、高效率、易用和灵活等优点. 本文将在文献[2]的基础上，设计、实现基于 MFC 的 AES 可视化试验平台，以期形象地再现 AES 算法加/解密的迭代过程.

1　高级加密标准 AES

AES，即高级加密标准. 2000 年 10 月，NIST 宣布通过对 15 种候选算法进行比较，选出 Rijndael 成为将来的 AES，作为新的密钥加密标准推广. 2001 年 11 月 26 日，NIST 正式公布高级加密标准 AES.

AES 是 NIST 针对电子数据的加密所制定的规范，后成为一种公认的可用来保护电子数据(包括财务数据、电信数据和政府数据)的新型加密算法，可以使用 128，192 和 256 位长的对称密钥块密码完成对 128 b(16 B)数据块的加密和解密，加密算法的输入和解密算法的输出均为 128 b. 图 1 是 AES 的加密与解密结构.

AES 算法的入口参数有 2 个，即 Key 和 Data，其中 Key 是 AES 算法的工作密钥，Data 是被加/解密的数据. 在通信网络的两端，双方约定一致的 Key，在通信的源点用 Key 对核心数据进行加密，然后以密文形式在公共通信网中传输到目的地后，用同样的 Key 对密文进行解密，便可还原出明文形式的核心数据. 这样就保证了核心数据在公共网络中传输的安全性和可靠性. 通过定期同时更新通信双方的 Key，便能提高数据的保密性.

1.1　AES 密钥

Rijndael 算法具有如下特性：1)对所有已知的攻击具有免疫性；2)在各种平台上，其执行速度快而

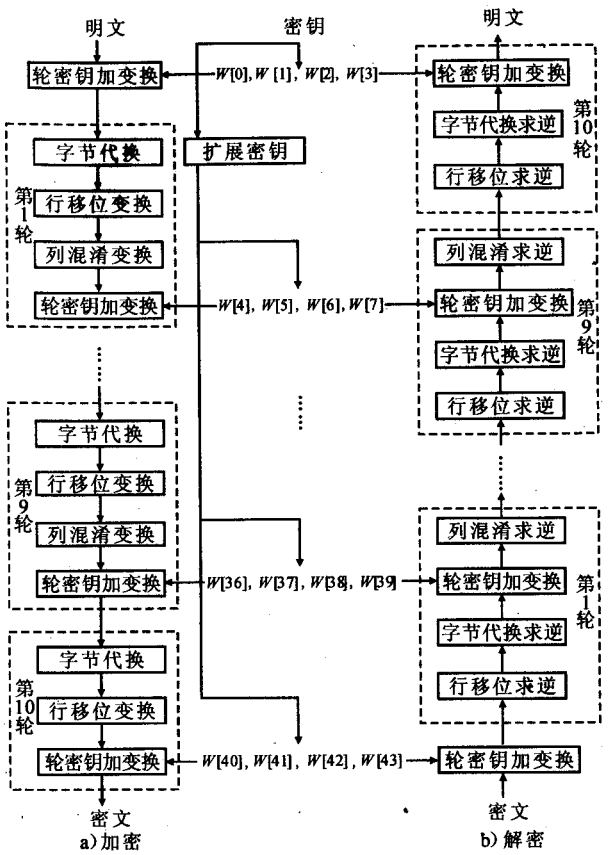

图 1　AES 的加密与解密结构

且代码紧凑.[3]在 Rijndeal 算法中，分组长度和密钥长度均能分别指定为 128 b，192 b 或 256 b. 在高级加密标准规范中，密钥的长度可以使用三者中任意 1 种，但分组长度只能是 128 b，本设计使用的密钥长度即为 128 b.

1.2　Rijndael 的步骤

与 DES 的工作方式类似，Rijndael 也使用了替换和置换的基本操作，由密钥长度决定需要运行的轮数，最少的轮数为 10(当密钥长度为 128 b 时)，最多的轮数是 14，但是 Rijndael 操作都涉及到整个字节，而不是字节中的位.

1.2.1　一次性的初始化处理　1)扩展 16 B 的密钥. 与正常情况下一样，该算法的输入是密钥和明文，但此时密钥长度是 16 B，该步骤要将这个 16 B 密钥扩展到 11 个数组中，每个数组含有 4 行 4 列. 扩展后的密钥长度为 176 B，即初始 16 B 的密钥被扩展为一个长度为 176 B 的密钥，这 11 个密钥数组(标号为 W_0—W_{43})中，1 个用于初始化处理，其余 10 个用于随后的 10 轮操作. 密钥的扩展过程：首先将 16 B 的初始密钥复制到扩展密钥块的第 1 个数组中(标号为 $W_0 \sim W_3$)；随后逐个填充剩余的 10 个数组(标号为 $W_4 \sim W_{43}$)，在填充数组 $W[i]$ 时，若索引

值 i 是4的倍数,对该数组的填充使用如下所示的复杂逻辑,对于其他的数组只使用简单的XOR运算即可.

```
ExpandKey(byte K[16],word W[44])
{
  word tmp;
  for(i=0;i<4;i++)
    {W[1]=K[4*i],K[4*i+1],K[4*i+2],
K[4*i+3];}
  for(i=4;i<44;i++)
  {tmp=w[i-1];
    if(i mod 4==0)
      tmp=substitutr(Rotate(tmp)XORConstant[i/4];
    w[i]=w[i-1]XOR tmp;
  }
}
```

密钥扩展的复杂性和AES其他阶段的复杂性确保该算法的安全性.密钥扩展过程中各数组的详细变化过程可参见文献[4].

2)扩展16 B的明文块.把16 B的明文块复制到一个称为State的二维4×4数组中,复制顺序按列进行,即将该明文块的前4 B复制到State数组的第1列中,接下来的4 B复制到State数组的第2列中,以此类推.

3)State数组与密钥进行XOR运算.扩展密钥的前16 B(即 $W_{[0]}$,$W_{[1]}$,$W_{[2]}$ 和 $W_{[3]}$)分别与State数组中对应位置的字节进行XOR运算,于是State数组中的每个字节被其自身与扩展密钥的对应字节进行XOR运算后的结果所替换.

明文块的扩展过程以及State数组与密钥块间的XOR操作可参见文献[4].

1.2.2 轮处理操作 1)字节代替变换.AES定义了1个 S 盒,它是由16×16个字节组成的矩阵,包含了8 b值所能表达256种可能的变换.State数组中的每个字节按照如下的方式映射为1个字节:把该字节的高4 b作为行值,低4 b作为列值,然后取出 S 盒中对应行列的元素作为输出.注意,这里AES使用1个 S 盒,而DES使用的是3个 S 盒.

S 的结构及构造方式可参见文献[4].S 盒的每个字节映射为它在有限域 $GF(2^8)$ 中的逆,"00"映射为它本身;S 盒的构造过程中使用XOR(异或)和mod(取余)等运算操作.经过精心设计的 S 盒可以防止已有的各种密码分析的攻击,同时Rijndael的开发者特别寻求在输入值和输出值之间几乎没有相关性的设计,且输出值不是通过简单的数学函数变换输入值而得到.当然出于解密过程的考虑,S 盒必须是可逆的,即 S 盒[S 盒(a)]=a;然而,S 盒(a)=逆 S 盒(a)不成立,即 S 盒不是自逆的.

2)行移位变换.将State数组的第1行保持不变,把State数组的第2行循环左移1个字节,State数组的第3行循环左移2个字节,State数组的第4行循环左移3个字节,其变换过程如图2所示.

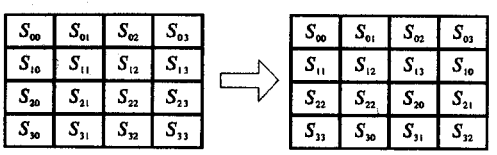

图2 行移位变换

在加密过程中,明文的前4 B直接被复制到State数组的第1列中,接着的4 B被复制到State数组的第2列中……而轮密钥也是逐列应用到State上的,因此行移位就是将某个字节从一列移到另一例中,移动的线性距离是4 B的倍数.

3)列混淆变换.每列中的每个字节被映射为一个新值,此值由该列中的4 B通过函数变换得到,这个变换可以由基于State的矩阵乘法表示,如图3所示.乘积矩阵中的每个元素是一行和一列中所对应元素的乘积之和,在这里的乘法和加法都是定义在有限域 $GF(2^8)$ 上的.

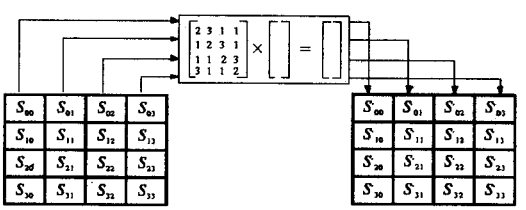

图3 列混淆变换

4)轮密钥加变换.128 b的State按位与128 b的轮密钥进行XOR运算,即该操作是基于State列的操作.把State的一列中的4 B与轮密钥的1 B进行XOR运算,可将其视为字节级别的操作,变换过程如图4所示,定义为 $E=AR(D)$.高级加密标准AES的加解密过程可参见文献[4-5].

1.3 AES算法安全性描述

从应征为侯选算法,到被选为终选算法,到最后

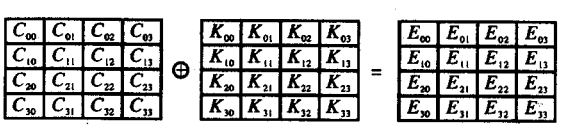

图4 轮密钥加变换

被确定为AES标准,直至现在,Rijndeal算法一直经受着世界各国密码机构和专家的强烈攻击.但是,目前尚未有对完整Rijndeal算法的成功攻击,只有几种对减少轮数的简化算法的攻击方法.最有名的当数密码设计者自己提出的Square攻击,其主要思想是利用第4轮字节替换前后平衡性的改变来猜测密钥字节,对128 b密钥下4—6轮简化算法有效.[6]

2　试验平台功能模块设计

AES算法的加/解密过程较为繁琐,为信息安全的教学带来诸多不变,因此有必要设计可视化演示软件将其复杂的迭代过程形象地展现出来,便于理解和学习.

根据AES算法的加/解密过程,基于MFC的AES可视化试验平台包含如图5所示的模块.

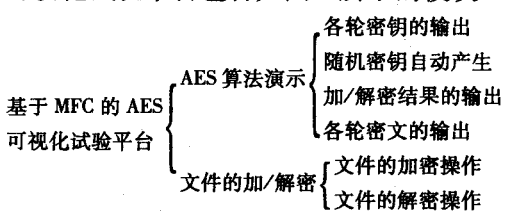

图5　演示平台功能模块设计

该试验平台主要有2类功能模块——AES算法演示模块和文件的加/解密模块.其中AES算法演示模块是将输入的16 B明文在密钥的作用下经过一系列的变换转化为密文;同样,得到的密文在密钥的作用下可以解密还原为原始明文,解密过程即是加密的逆过程.在该模块中动态显示AES算法在各轮迭代中密钥和密文的变化情况,文件的加/解密模块完成对文件的加解密操作.

3　技术实现

3.1　AES算法演示模块

3.1.1　密钥的产生　加密模块对输入平台的明文进行加密,产生出密文,这一过程中需要密钥与密文进行一定轮次的相关操作,而该密钥可以由用户自主输入,也可以由平台随机产生.平台可以随机产生3种类型的随机密钥(第1类随机密钥由数字字母组成;第2类由汉字组成;第3类由数字、字母和汉字组成).

本文试验平台在随机产生汉字密钥时使用由1 024个汉字组成的汉字密钥库(data.dat),平台每次随机到库文件中读取1个汉字,由于1个汉字占2 B,所以由汉字组成密钥时,密钥的长度选择偶数.

3.1.2　加密过程　加密过程要求输入16 B的明文数据,首先对该明文数据通过初始化扩展成为二维数组,在密钥控制下对输入的明文进行相应轮次的迭代形成密文数据.因为这一过程对所有传输的数据都进行了加密,所以加密过程对用户是透明的.试验平台加密算法的主要组成部分如下.

```
void Encryption(unsigned char * input, unsigned char * output)
{……
  for(int t=0;i<4*Nb;i++)//16 B明文块的初始化扩展
    State[i%4][i/4]=iput[i];
  AddRoundKey(0);//轮密钥加变换
  for(int round=l;round<=(Nr-1);round++)
  {//主循环,Nr为相应长度的密钥对应的轮数
    SubBytes();//字节代换
    ShiftRows();//行移位变换
    XixColumns();//列混淆变换
    AddRoundKey(round);//轮密钥加变换
  }
  SubBytes();//字节代换
  ShiftRows();//行移位变换
  XixColumns();//列混淆变换
  AddRoundKey(Nr);//轮密钥加变换
  ……
}
```

3.1.3　解密过程　如图1所示,AES的解密算法与加密算法不同,尽管在加密和解密过程中密钥的扩展形式一样,但在解密中的变换顺序与加密中的变换顺序不同.其缺点在于对于同时需要加密和解密的应用而言,需要2个不同的软件或固件模块分别完成.试验平台的解密算法的主要组成部分如下.

```
Void Decrypt(unsigned char * input,unsigned char * output)
{……//完成相应的准备操作
  AddRoundKey(Nr);//轮密钥加变换
  for(int round=Nr-1;round>=1;round--)
  {//主循环
    InvShiftRows();//行移位变换逆过程
    InvsubBytes();//字节代换逆过程
    AddRoundKey(round);//轮密钥加变换
    InvMixColumns();//列混淆变换逆过程
  }
  InvShiftRows();//行移位变换逆过程
  InvSubBytes();//字节代换逆过程
  AddRoundKey(0);//轮密钥加变换
```

```
InvMixColumns( );//列混淆变换逆过程
……//对还原后的明文进行变换后输出
}
```

由于逆向行移位影响State中字节的顺序,但不更改字节的内容,同时也不依赖字节的内容来完成它的移位. 而逆向字节代换影响State中的字节内容,但不更改字节的内容,也不依赖字节的顺序来进行它的代换,因此这2个操作可以互换.

轮密钥加和逆向列混淆并不更改State中字节的顺序,这2个操作对列输入是线性的,即对给定的State和给定的轮密钥 W_j 恒有公式①成立:

$$\begin{aligned}&\mathrm{InvMixColumns}(\mathrm{State}\oplus W_j)=\\&[\ \mathrm{InvMixColumns}(\mathrm{State})\ ]\oplus\\&[\ \mathrm{InvMixColumns}(W_j)\ ]\end{aligned}\qquad①$$

下面用线性代数的知识证明公式①的成立,假定State的第1列是(Y_0,Y_1,Y_2,Y_3)和轮密钥 W_j 的第1列为(K_0,K_1,K_2,K_3),则有

$$\begin{pmatrix}E&B&D&9\\9&E&B&D\\D&9&E&B\\B&D&9&E\end{pmatrix}\begin{pmatrix}Y_0\oplus K_0\\Y_1\oplus K_1\\Y_2\oplus K_2\\Y_3\oplus K_3\end{pmatrix}=$$

$$\begin{pmatrix}E&B&D&9\\9&E&B&D\\D&9&E&B\\B&D&9&E\end{pmatrix}\begin{pmatrix}Y_0\\Y_1\\Y_2\\Y_3\end{pmatrix}\oplus\begin{pmatrix}E&B&D&9\\9&E&B&D\\D&9&E&B\\B&D&9&E\end{pmatrix}\begin{pmatrix}K_0\\K_1\\K_2\\K_3\end{pmatrix}$$

以第1行为例,有

$$\begin{aligned}&[\{E\}\cdot(Y_0\oplus K_0)]\oplus[\{B\}\cdot(Y_1\oplus K_1)]\oplus\\&[\{D\}\cdot(Y_2\oplus K_2)]\oplus[\{9\}\cdot(Y_3\oplus K_3)]=\\&[\{E\}\cdot Y_0]\oplus[\{B\}\cdot Y_1]\oplus[\{D\}\cdot Y_2]\oplus\\&[\{9\}\cdot Y_3]\oplus[\{E\}\cdot K_0]\oplus\\&[\{B\}\cdot K_1]\oplus[\{D\}\cdot K_2]\oplus[\{9\}\cdot K_3]\end{aligned}\qquad②$$

方程②的成立证明了公式①的成立,因此,若先对轮密钥应用逆向列混淆,可以变换解密算法中的轮密钥加和逆向列混淆操作.

在解密轮中前2个阶段交换、后2个阶段交换后就将使解密算法的结构具有与加密算法相同的结构,而改变后的结构与原解密函数等价,等价的解密函数结构如图6所示.

该试验平台使用等价的解密算法同样可以对加密后的明文进行正确无误的解密,将其还原为初始明文.

3.2 主循环操作

AES每一轮的操作都是由4种不同的变换组成,

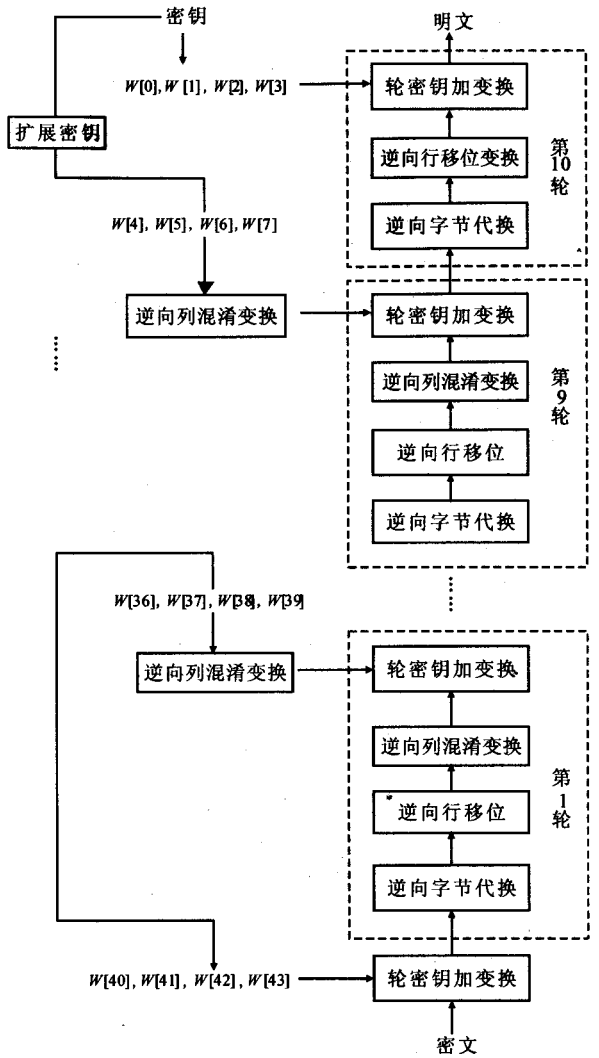

图6 等价的解密算法结构

这些操作依次为字节代换、行移位、列混淆和轮密钥加. 轮密钥加实质是一种Vernam密码形式,另外3个阶段一起提供混淆、扩散以及非线性功能. 这种方式非常有效且安全级别高,每个阶段均可逆,对字节代换、行移位和列混淆,在解密算法中用与其相对应的逆函数,而轮密钥加的逆过程就是轮密钥和分组相异或.

4 平台功能测试

分别对该可视化试验平台的AES算法演示模块和文件的加/解密效果进行测试,结果如下.

4.1 加密过程演示测试

本文使用系统随即产生的密钥来对输入的明文进行加密效果测试,点击加密界面的“密钥自动生成”按钮产生随机密钥 NaviLJq5zpgFJjBljbOifax. 同时输入16 B长的明文:厚德积学励志笃行,密钥

长度选择 192 b，如图 7 所示，可以查看 12 轮变换过程中每一轮密钥和密文的具体值. 点击“最终加密结果”按钮，平台显示出最终的加密密文为 3d1c4b161099bba05ad8e84ba5c2b24c，同时平台显示出加密过程中密文和密钥的变化情况.

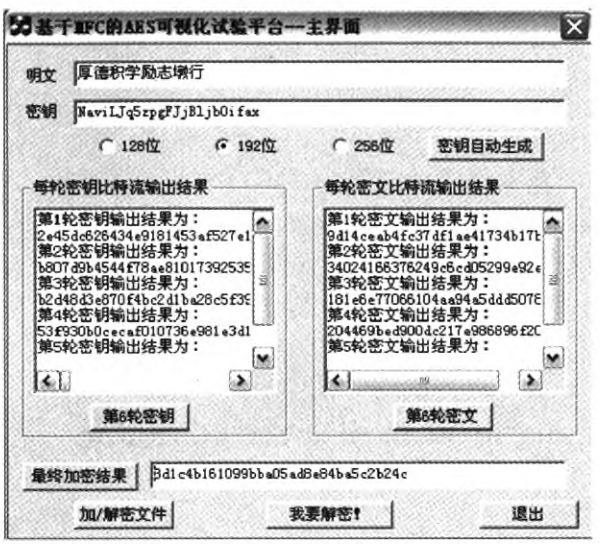

图 7　明文加密

4.2　解密过程演示测试

加密过程结束后，记住加密过程所使用的密钥和加密后的密文，点击“我要解密”按钮，点击该按钮后，主界面下方将显示解密模块界面. 在解密模块的界面上相应的文本框中输入密钥与密文，点击“解密”按钮，平台显示出与密文相对应的原始明文：厚德积学励志笃行.

4.3　文件加密效果测试

该平台同时实现了对文件的加/解密，图 8 为文件的加/解密过程显示. 在对文件进行加密之前牢记初始文件的格式，试验平台加密后的密文格式并非与明文格式相同，所以在解密加密文件时还需输入初始明文的原文件格式. 注意待加/解密的文件大小必须超过 16 B.

5　结语

本文开发了基于 MFC 的 AES 可视化试验平台.

图 8　文件的加解密

该平台可方便观测 AES 加密操作中各阶段密文和密钥的变化过程，为网络安全教学中 AES 可视化演示提供了技术支持. 此外，该平台还可以用于对秘密文件的加密，从而为重要数据提供高级别的安全防护.

参考文献：

[1] Claessens J, Diaz C, Goemans C. Revocable anonymous access to the internet[J]. Int Research Electr Networking Appli and Policy, 2003, 13(4): 13.

[2] 周彦伟，吴振强. 基于 MFC 的 DES 算法演示平台的设计与实现[J]. 电子技术应用，2009，35(5)：145.

[3] 许茂智，游林. 信息安全与密码学[M]. 北京：清华大学出版社，2007：61－70.

[4] William Stallings. 密码编码学与网络安全——原理与实践[M]. 4 版. 孟庆树，王丽娜，傅建明，等译. 北京：电子工业出版社，2008：96－124.

[5] Altu Kahate. 密码学与网络安全[M]. 2 版. 北京：清华大学出版社，2009：108－117.

[6] 韦宝典. 高级加密标准 AES 中若干问题的研究[D]. 西安：西安电子科技大学，2003.

基于MFC的AES可视化试验平台的设计与实现

作者： 乔子芮， 周彦伟， QIAO Zi-rui， ZHOU Yan-wei

作者单位： 乔子芮,QIAO Zi-rui(陕西师范大学计算机科学学院,陕西西安,710062)， 周彦伟,ZHOU Yan-wei(陕西师范大学教师专业能力发展中心,陕西西安,710062)

刊名： 郑州轻工业学院学报（自然科学版） ISTIC

英文刊名： Journal of Zhengzhou University of Light Industry(Natural Science Edition)

年，卷(期)： 2011,26(6)

参考文献(6条)

1. Claessens J;Diaz C;Goemans C Revocable anonymous access to the i ntemet 2003(04)

2. 周彦伟,吴振强 基于MFC的DES算法演示平台的设计与实现[期刊论文]-电子技术应用 2009(5)

3. 许茂智;游林 信息安全与密码学 2007

4. William Stallings;孟庆树;王丽娜;傅建明 密码编码学与网络安全——原理与实践 2008

5. Altu Kahate 密码学与网络安全 2009

6. 韦宝典 高级加密标准AES中若干问题的研究[学位论文] 2003