基于 X-IDEA 算法实现手机通信中图文加密①

肖四友

(浙江万里学院 智能控制研究所，宁波 315100)

摘 要：研究手机通信系统中图文数据的加密设计，由于手机设备的有限性能，其运算能力还无法与计算机相提并论。因此文中采用具有高效率和轻量级特点的 X-IDEA 加密算法，它是对 IDEA 算法进行改良使之更适合运用于手机图文安全传送，也更适合与当前 3G 通信中图文大小可变以及存在相同部分的可能性增大的情况。

关键词：手机通信；图文加密；IDEA 算法；X-IDEA 算法

Image-Text Encryption Techniques in Mobile Communications Based on X-IDEA

XIAO Si-You

(Intelligent Control Research Institution, Zhejiang Wanli University, Ningbo 315100, China)

Abstract: This paper has studied the encryption of data in mobile's graphic communication, and the computing power cannot be compared with the computer due to the limited performance of mobile devices. Therefore, this paper uses a highly efficient and lightweight characteristics: X-IDEA encryption algorithm, it is a improved IDEA algorithm. This nalgorithm has improved IDEA more suitable transmission used in mobile graphics safe, more suitable the current variable size Photo in 3G, while the presence of an increased possibility of the same part of the case.

Keywords: mobile communication; image-text encryption; IDEA algorithm; X-IDEA algorithm

随着国内 3G 推广，手机应用的领域进一步扩大，即时通信和图文传输等更加方便。传输数据进行加密处理以保证通信数据的安全性变得尤为重要，特别在于手机用户通信的过程中进行图文传输，既要考虑传输质量，更要考虑传输安全，因此必须采取必要的加密操作[1]。但受限于手机设备的有限性能，其运算能力还无法与计算机相提并论，因此其加密技术更加值得大家去努力突破。

1 手机加解密概述

1973 年 IBM 公司设计了 DES(Data Encryption Standard)，多年来一直作为非常安全的加密算法被用于各种数据的保护，但随着科技的进步，DES 算法也因密钥长度短(56 位)、密码空间小，现有的计算机的采用穷举法进行破解来获取正确密钥已趋于可行。同样，传统 64 位 GSM 手机网络加密技术于 2009 年被 28 岁的德国电脑工程师 Karsten Noh 公开破解，任何人都可以利用破解出来的密码本破译截获的手机信息。

而 IDEA(Internation Data Encryption Algorithm)数据加密算法是由中国的来学嘉博士和著名的密码专家 James L.Massey 于 1990 年联合提出的，该算法用硬件和软件实现都很容易，而且比 DES 在实现上快的多。IDEA 也被认为是目前世界上最好最安全的分组密码算法，且对计算功能要求不高，所以可以方便地使用在手机中。IDEA 的密钥长度是 128 位，在穷举攻击的情况下，IDEA 需要经过 2128 次加密才能恢复出密钥，假设芯片每秒能检测 100 亿个密钥，需要 10 年，故其安全性很高[2]。

2 IDEA数据加密算法

IDEA 数据加密算法是对 64bit 大小的数据块加密的分组加密算法，它是基于“相异代数群上的混合运

① 基金项目:“核.高.基”专项项目(2009ZX01039-001-002-004)
收稿时间:2010-07-19;收到修改稿时间:2010-09-18

© 中国科学院软件研究所 http://www.c-s-a.org.cn

算”的设计思想，该算法既可用于加密，又可用于解密[3]。

2.1 子密钥的生成

IDEA 算法加密和解密都各自需要 52 个子密钥，每一个子密钥 16 位，由 128 位密钥生成。加密子密钥的生成过程是：对 128 位密钥进行分组处理，分成 8 组，每组 16 位，从而可以得到 8 个加密子密钥，将前面生成的 8 个加密子密钥 128 位循环左移 25 位，又做一次 16 位分组，从而又可以得到 8 个加密子密钥，如此循环，将这个过程循环 7 次，在第 7 次的循环中，只需取前面的 4 个加密子密钥，至此便产生了全部的 52 个加密子密钥。

解密子密钥的生成过程是：首先根据密钥生成加密子密钥，然后对加密子密钥进行换位操作，最后对换位操作后的加密子密钥进行部分子密钥的模 216 的加密逆运算和剩余部分子密钥的模(216+l)的乘法逆运算。

2.2 加密解密过程

IDEA 算法中，明文长度为 64 位。加密过程是：对明文进行分组处理，分为 4 组：X1、X2、X3、X4，每组 16 位。X1、X2、X3、X4 作为第一轮的输入，在每一轮中，4 个输入分组与 6 个 16 位加密子密钥分别做模 216 的加法、模(216+l)的乘法和异或操作，得到四个输出结果，将其又作为下一轮的输入。如此循环进行八轮操作。将最后得到的结果再与 4 个加密子密钥作输出变化，从而得到最后的加密密文。

解密过程是加密过程的逆，在解密的过程中只需把加密子密钥替换为解密子密钥就可以实现对加密后的密文进行解密。

3 X-IDEA应用于手机的图文加解密

IDEA 算法中，明文的长度固定且比较短只有 64 位，而在手机通信系统上对于通信的消息中相同的部分会被加密成相同的密文，从而暴露了明文的数据格式和某些统计学特性，降低了明文的保密性。根据即时通信传输的消息的长度不一定性和消息存在相同部分可能性比较大的特点，因此本系统提出对 IDEA 算法进行改进，即 X-IDEA 算法，使之更适合运用于手机通信系统中图文数据的加解密处理，同时改进的算法经过测算其被穷举突破在目前的技术和普通计算机的计算能力下当前基本无法实现，所以该算法对于解决手机图文通信的安全是有保障的。

3.1 X-IDEA 加密算法设计

本系统设计的 X-IDEA 算法，使之更适合运用于手机即时通信系统中图文的加解密。它主要从明文的长度、加密过程和解密过程三个部分对传统的 IDEA 算法做了改进。X-IDEA 算法实现对手机图文传输中加解密的流程如图 1 所示。

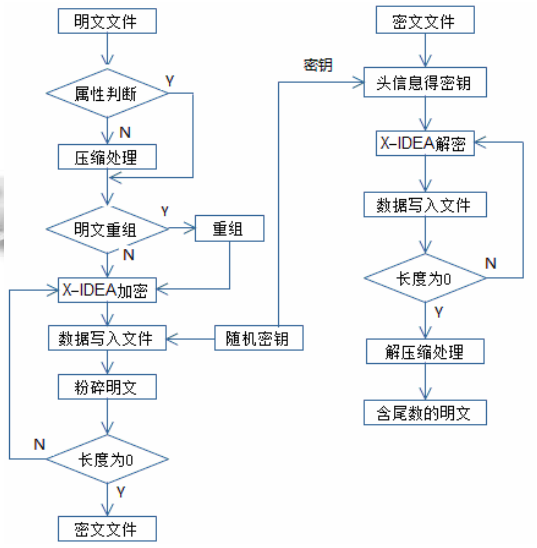

图 1　X-IDEA 流程图

3.1.1 明文长度

X-IDEA 算法中，明文的长度不再做限制，明文的分组长度为 64 位。因为明文的长度不做限制，因此对明文进行加密前，需要对明文进行重组判断，如果明文的长度不是 64 的整数倍，将对明文进行重组，在明文末尾加上一些特殊字符，使明文的长度达到 64 的整数倍，如果明文的长度是 64 的整数倍，则不做重组处理。

3.1.2 文件加密过程

加密前先根据文件的大小判断是否对文件进行压缩，然后获取文件的属性，并连同从数码锁中产生的密钥随机索引一同保存于文件的头信息中，然后创建一个临时文件，并将文件头所包含的信息写入其中。接下来判断文件长度是否为 64 的倍数，若是则按照分组加密的方法进行加密，否则在文件尾部填充随机函数产生的随机数 X，使得文件的长度刚好是 64 的倍数，之后再进行分组加密，并把加密结果写入临时文件，删除原文件后将临时文件命名为原文件。这样的输入输出机制保证了文件的属性和文件名在加密前后的一

© 中国科学院软件研究所 http://www.c-s-a.org.cn

致性。

3.1.3 文件解密过程

解密时首先从数据块中读出文件头信息，根据头信息中的密钥索引取出加密密钥得到加密子密钥，再由解密模块的对应关系求得解密密钥，然后分块解密密文，解密后的数据中包括原文件数据和填充到原文件尾部的随机数，根据文件头信息中保存的文件实际长度，恢复原文件明文。

X-IDEA 算法推理如下[4]：

加密过程

$$\begin{cases} C_1 = IDEA(C_0) \oplus P_1 \\ C_i = IDEA(P_{(i-1)}) \oplus P_i (1 < i \le n) \end{cases}$$

$$\begin{cases} P_1 = IDEA(C_0) \oplus C_1 \\ P_i = IDEA(P_{(i-1)}) \oplus C_i (1 < i \le n) \end{cases}$$

从上述方程可以计算出：

$$P1 = IDEA(C_0) \oplus C_1 = IDEA(C_0) \oplus IDEA(C_0) \oplus P_1 = P_1$$

$$P_i = IDEA(P_{(i-1)}) \oplus C_i = IDEA(P_{(i-1)}) \oplus IDEA(P_{(i-1)}) \oplus P_i$$

因此，明文经过加密，再经过解密，产生加密前的明文，说明算法正确。

3.2 X-IDEA 加解密实现

3.2.1 加密过程

加密的过程是对重组后的明文进行分组处理，每组 64 位。由随机数产生器随机产生数据 A，作为加密的初始数据，A 为 64 位，运用 IDEA 数据加密算法对数据 A 和除最后一个明文分组之外的所有明文分组进行加密，产生密文 C_0 和密文分组 D_1，D_2，$D_3 \cdots D_{n-1}$，对密文 C_0 再次进行加密操作，将产生的密文 C0’与明文 P_1 进行异或操作 $C_1=C_0' \oplus P_1$ 产生密文 C_1，再将密文 D_1 与明文 P_2 进行异或操作产生密文 C_2，密文 D_2 与与明文 P_3 进行异或操作产生密文 C_3，依次方式进行运算产生直到所有的密文 $C=C_1C_2C_3 \cdots C_n$。最后将密文 C_0 置于密文 C 头部形成总的密文 $C=C_0C_1C_2 \cdots C_n$，一起发送给目标用户。具体的算法描述如下：

//加密算法

输入：待加密的明文 P 和密钥 E_k

输出：加密后的密文 C

① X←randomX(); //由随机数生成器生成 64 位数据 X

② P←reform_Data(P); //对明文进行重组处理，如果明文不是 64 的整数倍，则在明文字节数组后面添加特殊字符，直到满足 64 的整数倍，如果是 64 的整数倍，则不做任何处理

③ Array←group(P); //重组后的明文进行分组处理

④ C_0←encrypt(E_k,X); //对数据 X 使用 IDEA 数据加密算法进行加密

⑤ add(C_0,C); //将 C_0 添加到结果集 C 中

⑥ C_0’ ← encrypt(Ek,X);

⑦ for each P_i in Array

⑧ $C_i \leftarrow C_0' \oplus P_i$; //异或操作

⑨ add(C_i,C); //将 C_i 添加到结果集 C 中

⑩ if P_i 不是最后一个分组

⑪ C_0’ ← encrypt(E_k, P_i);

⑫ end for

⑬ return C; //返回加密后的密文

⑭ 算法结束

3.2.2 解密过程

解密的过程是对密文进行分组处理，分为 $C_0,C_1,C_2,C_3,\cdots C_n$(n+l)个组，每组也是 64 位。对密文的第一个分组 C_0 使用 IDEA 数据加密算法进行加密，产生密文 C_0’，并将产生的密文 C_0’与密文 C_1 进行异或操作，产生明文 P_1，对明文 P_1 进行加密操作，将产生的密文与密文 C_2 进行异或操作，产生明文 P_2，对明文 P_2 进行加密操作，将产生的密文与 C_3 进行异或操作，产生明文 P_3，依次方式进行运算，直到产生所有的明文 $P=P_1P_2P_3 \cdots P_n$。最后对产生的明文进行重组操作，去除在明文加密前重组时加入的特殊字符。具体的算法描述如下：

//解密算法

输入：待解密的密文 C 和密钥 E_k

输出：解密后的明文 P

① Array←group(C); //对密文进行分组处理

② C_0’ ← encrypt(E_k, C_0); //对密文的第一个分组使用 IDEA 数据加密算法进行加密处理

③ for each C_i in Array

④ $P_i \leftarrow C_0' \oplus C_i$; //异或操作

⑤ add(P_i,P); //将 P_i 添加到结果集 P 中

© 中国科学院软件研究所 http://www.c-s-a.org.cn

⑥ C_0’ ← encrypt(E_k, P_i);

⑦ end for

⑧ P← reform_data(P);　//对解密后的明文进行重组，去掉添加的特殊字符

⑨ retrun P;

⑩ 算法结束

3.3 加解密的具体实现

根据 X-IDEA 算法进行的加密程序设计如图 2 所示，C:\t1.jpg 是加密源文件，而 tt1.jpg 是加密后的源文件，通过图 2 的加密后可以发现 tt1.jpg（图 3 所示）已经被加密而无法查看其原内容，而使用该加解密程序进行对应的解密程序可以还原 tt1.jpg 到 t1.jpg，由此可以看出该算法的加解密过程是可以实现的，并且在 Android 基础上进行了验证，完成了手机通信中的图文加解密技术。

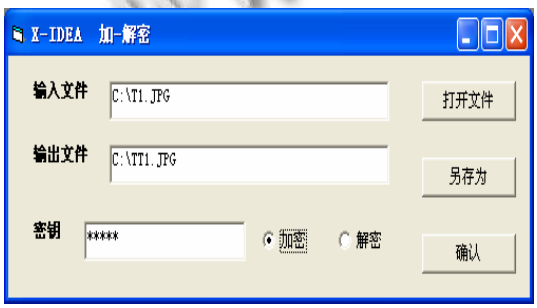

图 2　加解密应用程序

图 3　TT1.jpg 原图

4　结论

本文首先对手机即时通信系统进行了介绍，然后着重介绍了该系统中图文的加密算法。先对 IDEA 进行了简单的介绍，然后结合手机通信的特点，对 IDEA 算法进行了改进，提出了 X-IDEA 算法原理，并设计了该算法的实现流程，最后设计了加解密程序实现对图片的加解密，达到了确保图片文件传输的安全，随后将之引入到手机上运行也得到了预期目标。

参考文献

1 公磊,周聪.基于 Android 的移动终端应用程序开发与研究.计算机与现代化,2008,(8):85–89.

2 贾琴勇.即时通讯系统的研究与实现[硕士学位论文].武汉:武汉理工大学,2008.28–36.

3 潘凤,王华军,苗放,李刚.基于XMPP协议和Openfire的即时通信系统的开发.计算机时代,2008,(3):15–19.

4 张海燕.Java 多线程技术在手机互联网中的应用.农业网络信息,2008,(3):97–98.

（上接第 247 页）

推广中具有一定的意义。

参考文献

1 唐荣锡,汪嘉业.计算机图形学教程.北京:科学出版社,1990.60–65.

2 Bresenham JE. Algorithm for computer control of a digital plotter. IBM Systems Journal, 1965,4(1):25–30.

3 Rokne J, Rao Y. Double-Step incremental linear interpolation. ACM Transactions on Graphics, 1992,11(2):183–192.

4 刘勇奎.一个对称的快速直线生成算法.微计算机应用,1993,14(2):42–43.

5 Foley JD. Introduction to Computer Graphics. Beijing: China Machine Press, 2004:48–56.

6 贾银亮,张焕春,经亚枝.Bresenham 直线生成算法的改进.中国图象图形学报,2008,13(1):158-161.

7 郑宏珍,赵辉.改进的 Bresenham 直线生成算法.中国图象图形学报,1999,7(14):606-608.

8 苗兰芳,刘新国.自适应多步位移码直线绘制算法.软件学报,2002,(13)4:637–642.

9 刘晶,李俊,孙涵.改进的Bresenham直线生成算法.计算机应用与软件,2008,25(10):67–71.

10 孙岩,唐棣.并行的 Bresenham 直线生成算法.计算机工程与应用,2001,37(21):37–42.

11 Yao C, Rokne JG. Bi-Directional incremental linear interpolation. Computer & Graphics, 1996,20(2):295–305.

12 宣淑巍,李晓江,马成炎.一种基于循环减法原理除法器的加速方法.微电子学与计算机,2009,26(12):12–15.

© 中国科学院软件研究所 http://www.c-s-a.org.cn