# 使用双密钥的实时加密

摘 要

这篇文章介绍了一种提高在暴力破解攻击下保护重要数据的加密方法。这种方法是使用两组不同的密钥同时对数据进行加密，一个普通密钥，另一个是基于时间的密钥。加密算法使用改进过的传统的加密方法，但是解密时有一点复杂，需要检查两个密钥的正确性和相关时间的有效性。这克服了使用暴力方法对两个连续的密钥进行分析完全取决于系统的速度的攻击。同时，本文还提出了动态选择加密的轮数的方案，增加了密码分析的难度。该系统要求 只能在单一系统和网络下进行解密。通过C语言实现，并进行密码分析，以检查保护等级。

**关键字**：加密，解密，实时系统，基于时间的密钥，暴力破解攻击，密码分析

# 1、介绍

众所周知，正是因为通过硬件实现软件的功能，嵌入式现场可编程逻辑阵列的使用及大地降低了加密算法的安全性。高水平的制造技术更重要的是并发性的数据处理可以通过暴力攻击找到安全漏洞。因此提出了多种加密方法对数据不断进行加密。一个方法就是使用组成椭圆曲线（ECC）的加密和数据加密标准（DES）[1]。该算法证明了DES是一种高效的算法，密钥会很容易被泄露。算法[2]是基于整数因数分解难度，并命名为基于矩阵的不对称批量加密算法。有一种新型的加密算法基于最佳字母树（OATs）的应用[3]，还有一种新型的叫做RAINBOW的面向流的流密码使用两个密钥，有别于真实密钥的时间密钥和真实密钥进行加密。[5]讨论了一种适合VLSI实施的加密算法。以上加密算法都是使用的是易受暴力破解攻击的技术。加密系统的安全性取决于加密消息的密钥大小，128位的密钥在现在可以提供最高的安全等级，但是在未来运算速度不断提高的情况下密钥的位数还需不断提高。此外，当用于暴力攻击的系统数量增加了系统之间共享的密钥空间时，安全级别再一次被削弱。总结一下，安全级别由一个系统的搜索密钥空间所需的时间决定。所提出的算法试图通过将时间作为关键的第二维度来克服这个缺陷。所以，时间为暴力攻击增加而纯粹不依赖于用于密码分析的系统的速度。这可以通过在解密中引入实时概念来实现处理。

# 2、传统加密算法

## 2.1 IDEA加密算法

现在的系统包含很多传统的加密算法，例如 国际数据加密算法（IDEA）和数据加密标准（DES）和BELOWFISH都是使用对称密钥的、面向块的加密算法。IDEA使用128位的密钥对64位的明文进行操作，使系统免疫于暴力破解的攻击。IDEA是基于一个基本功能，然后这样迭代八次。第一轮迭代在64位的输入明文上进行操作，然后剩下的连续操作就是在前一轮后的64位数据块上进行操作。在最后一轮迭代后，最终一步产生出64位的密文块。IDEA使用混淆和扩散的方法加密数据。64位的输入数据被分为四个16位的子块：X1、X2、X3、X4。这四个子块作为第一轮算法的输入。算法总共有八轮，每一轮中四个子密钥块相互和与16位子密钥进行异或、加、乘积运算。在每两轮运算之间第二和第三个子块之间要进行一次交换，最终，在八轮运算后的四个子块被收集起来组成输出变换的四个子密钥。

在每一轮中，每一轮的顺序如下：

1. X1乘第一个子密钥；
2. X2与第二个子密钥相加；
3. X3与第三个子密钥相加；
4. X4与第四个子密钥相乘；
5. 第一步与第三步的结果进行异或运算；
6. 第二步与第四步的结果进行异或运算；
7. 第五步的结果与第五个子密钥相乘；
8. 第六七步的结果相加；
9. 第八步的结果与第六个子密钥相乘；
10. 第七步与第九步的结果相加；
11. 第一步与第九步的结果进行异或运算；
12. 第三步与第九步的结果进行异或运算；
13. 第二步与第十步的结果进行异或运算；
14. 第四步与第十步的结果进行异或运算；

传统的算法受到以下缺点的影响：

现存系统的主要缺陷就是加密完全取只决于密钥，加密则只是执行预定轮数。现存系统在暴力破解下显得很脆弱，而且在合适的密码分析下很容易将消息内容导出。这是真的，因为在大多数情况下，在随机的“n”组密钥空间下，密钥可以在 “n/2”组的数据中被找见。

## 2.2 实时系统

实时系统能够在正确的时间中产生正确的输出，系统的内部处理，带来了系统控制的断续的结果，在任何时候，操作顺序是预先确定的。因此，用户可以在程序执行期间的任何时候都轻松地确定所有临时寄存器的值。如今，实时系统在需要处理速度非常高或者数据吞吐量非常大的各种领域才非常有用。实时系统输出的数据只有在预先定义的完整内部执行流程的预定时间内才可用。有很多实时操作系统（RTOS），诸如实时扩展（RTX 5.1）的操作系统的增强功能现在可用于在通用型台式计算机中实现实时操作。增加这样的设施使得实时系统甚至可以运行在桌面计算机中。这样的系统可以被用于分析所提出的系统，在这样的方式下输入从预先设定的时间间隔中得到反馈。这使得加密系统检测密钥的合法性和当前时间的间隔的合理性。

# 3、程序

## 3.1 加密

1. 128bit数据&128位密钥（64位数据&64bit时间）；

2. 产生随机数r作为轮数，每个数据组都进行操作；

3. 第一组数据使用可用n的函数操作r次；

4. 选择的函数顺序由生成的一组随机数决定；

例如：假设有4个可用的函数，随机数称为第一函数，称为第二函数，以此类推。

函数一：将32位数据倒置，结果与64位数据进行异或运算。

函数二：数据的奇数位与偶数位进行异或，反之亦然。

函数三：执行标准S盒代换，函数调用的顺序保存在日志中，日志也以相同的方式加密。

因此，密文由以下部分组成：

1. r——设置加密的轮数；
2. ABACBAB——函数调用的顺序函数一用A标识，以此类推（在这个条件下r=7）。这使得每组数据的操作的轮数是动态的；
3. 在r轮之后的结果作为密文。

既然，这个日志也是以与数据相同的方式进行加密的，只有一个密钥 – 也是密钥的一部分 - 将这些值通知给解密系统。 同样的过程继续下去，直到明文消失。

## 3.2 解密

解密是需要工作在实时系统的，当密钥的第一部分进入到解密系统时，系统将检查数据的有效性。然后启动一个定时器，它在64位时间表示的密钥的适当时间等待密钥的下一部分。对于有效时间间隔的有效密钥的每个条目都将使得程序继续进行解密。当所有的密钥都被在合适的时间送入系统，程序将结束，并给出解密后的明文。如果密钥和时间的有效性都丢失，处理将不再继续下去，所有的密钥都将从第一个输入。这个过程使每次尝试一个密钥都会消耗一定的时间，因为 时间的暴力攻击变得困难。

## 3.3 优点

1、在加入了时间因素之后，对加密分析的暴力破解将被有效组织。时间在每一小时和每一秒都将发生改变，可以是任何值，因此增加了密码分析的可能选择。

2、密钥和明文的大小在多种环境下都可以发生改变。

3、两个随机数，一个用于选择轮数，另一个用于选择函数调用的顺序。

4、这些时间都被记录下来，而且记录下来的登录数据也作为明文用同样的方法进行加密。解密时系统识别登录数据，然后 解密。

# 4、要求

通常，解密的程序需要在一台可控环境的独立的计算机上面完成。目标系统期望解密过程发生在单独的系统中，在时间的要求下，不可能通过网络进行解密，这对于系统的合法和非法用户都是如此。



图4.1 系统的执行流程（加密）



图4.2 系统执行的流程

加密过程不消耗时间，与现有的传统算法相似，解密过程将消耗时间 - 合法用户规定的最低限度和非法用户无限制。

# 5、对比分析

这部分将当前算法和传统算法的数学分析上的区别。假设如果密钥是10位的，那么总的密钥空间将有种组合，也就是1024种。每单位时间可以处理1次组合的系统将需要1024单位时间才能通过暴力攻击找出密钥。在另一方面，如果通过数据时间将密钥分为两个维度，那么将会出现下面的情况：



图5.1 目标算法 的时间分析

密钥空间有种组成，每个间隔的时间空间有种组成。可以根据输入能力以任何时间单位设定时间间隔。例如，如果时间以小时为单位，则每个时间间隔的最大间隔时间为32小时，如果时间为纳秒，则将提供32纳秒的时间间隔。但优势在于，在密码分析时的每一种情况下，时间消耗将会很高，从而使系统更加安全。总共的时间空间可以被计算成中密钥组合加4个时间间隔，结果是4个时间间隔乘32个单位时间所有的组合，即，2144种组合，因此安全性得到了提高。

# 6、结果与结论

这个系统已经使用C语言实现，而且使用可用的暴力破解进行攻击。在对比分析中分析了两组随机关键暴力攻击的系统响应。结果表明，每次强力攻击组合都需要花费大量的时间，因此很难破坏系统的安全性。所提出的方法增加了尝试进行密码分析的组合数量，每次尝试将消耗更多的时间，并且将使密码分析困难。

表6.1 时间分析的比较

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 算法 | 密钥大小 | 密钥搜索空间 | 时间搜索空间 | 密码分析的总时间 |
| IDEA | 10 | 1024 | - | 1024个单位时间 |
| 双密钥系统 | 10 | 32 |  | 2144个单位时间 |
| IDEA | 32 |  | - | 个单位时间 |
| 双密钥系统 | 32 | 65536 |  | 个单位时间 |

# 参 考 文 献

[1] Peng Gong au, Feng-jiao Qiu & Meng Liu, "A new algorithm based on DES and ECCfor CSCW", The 8th International Conference on Computer Supported Cooperative Work in Design, 2004. Proceedings. May 2004 Volume: 1 pp: 481 - 486.

[2] Mukesh Kumar Singh, "Matrix based asymmetric bulk encryption algorithm ", Information Assurance Workshop, 2004. Proceedings from the Fifth Annual IEEE SMC, June 2004, pp: 161 - 167.

[3] Arafat S M," An encryption algorithm based on alphabetic trees", The 3rd ACS/IEEE International Conference on Computer Systems and applications, 2005. pp 92.

[4] Ya-PintZhang, Jizhou Sun au and Xu Zhang, "A stream cipher algorithm based on conventional encryption techniques", Canadian Conference on Electrical and Computer Engineering, May 2004, pp: 649 - 652 Vol. 2.

[5] Foumaris A. P, SkIavos Nand Koufopavlou O,"VLSI architecture and FPGA implementation of ICE encryption algorithm", Proceedings of the 2003 10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003, Dec.2003, pp.88 – 91 Vol. 1

[6] Meena K, Dhanapal R, Janet B, "An Intelligent Information Retrival Agent", HC Research Journal, India, Vol 1, pp. 49 - 57, 2005.

[7] Atul Kahate, "Cryptography & Network Security ", Tata McGraw Hill, 2003.

[8] Bruce Schneier, "Applied Cryptography II Ed", Wiley Eastern, 1995.

[9] William Stallings, "Cryptography & Network Security", Prentice Hall, 1998.