第35卷 第23期 计 算 机 工 程 2009年12月
Vol.35 No.23 Computer Engineering December 2009
·安全技术· 文章编号：1000—3428(2009)23—0130—03 文献标识码：A 中图分类号：TP309

RC4流密码与微软Office文档安全分析

何克晶
(华南理工大学计算机科学与工程学院，广州 510641)

摘 要：根据微软官方文档、OpenOffice文档及wvWare实现等完全公开的信息，对RC4流密码及其在微软Office系列中的实现进行分析，认为Office 97~2003所默认使用的40 bit加密方式较不安全，通过结合Rainbow预计算攻击方法，证实其脆弱性。通过研究，建议不使用默认的“Office 97/2000兼容”40 bit加密，而采用更安全的“Microsoft Enhanced Cryptographic Provider”128 bit加密，或者使用压缩软件进行二次加密，从而进一步提高安全性。

关键词：RC4流密码；预计算攻击；微软Office；文档安全

Analysis of RC4 Stream Cipher and Microsoft Office Document Security

HE Ke-jing
(School of Computer Science and Engineering, South China University of Technology, Guangzhou 510641)

【Abstract】According to the open information from the Microsoft official documents, the OpenOffice documents and the wvWare project, this paper studies the RC4 stream cipher and its implementation in the Office 97~2003. The analysis discovers that the default 40 bit encryption method used by Office 97-2003 is very weak and insecure. Coupling rainbow precomputation attack, the encryption can be broken in 1 min~2 min. This paper suggests users do not rely on the default 40 bit “Office 97/2000 Compatible” encryption to protect your confidential information. On the contrary, the 128 bit “Microsoft Enhanced Cryptographic Provider” is preferred. It also recommends that users adopt the stronger encryption algorithm provided by compression softwares better when better security is necessary.

【Key words】RC4 stream cipher; precomputation attack; Microsoft Office; document security

1 概述

自从Ron Rivest提出RC4流加密方法以来，RC4就被广泛应用于多个领域，比如，用于保护IEEE 802.11无线网络安全的WEP(Wired Equivalent Privacy)，用于保护互联网传输层安全的TLS(Transport Layer Security)以及微软Office(Microsoft Office)等。一直以来，RC4被认为是种非常安全的流加密方法。但随着信息安全学家们发现RC4的最初若干字节密钥流的非均匀分布(非随机)特性[1]以及很多系统对RC4密码的非完美实现，使人们意识到基于RC4的加密系统并不像人们原先认为的那样安全[2]。比如，Borisov等人分析了WEP加密的安全性，利用RC4初始密码流的非均匀分布特性在几分钟内即能攻破WEP网络[3]。文献[4]分析了RC4在Microsoft Office中的错误使用，该错误使用同样的密钥流加密同一个文档的不同版本(即使该文档已被修改过)，且尽管RC4支持40 bit~256 bit的加密，因为出口的限制，在除美国外的许多其他国家所使用的RC4并不是128 bit的。比如，法国政府就曾经禁止民用超过40 bit的加密方法，而在许多国家的Microsoft Office中的RC4加密默认情况下也是40 bit的。随着计算机处理器和FPGA芯片技术的发展，即使是使用暴力破解法，使用分布式技术也可在数天之内破解40 bit的RC4[5]，而使用特制的FPGA芯片，该时间可缩短到几个小时[6]。

2 相关问题

Microsoft在其Office97~2003产品中使用RC4密码加密文档，其加密密码又分为“打开密码”和“修改密码”，如图1所示。

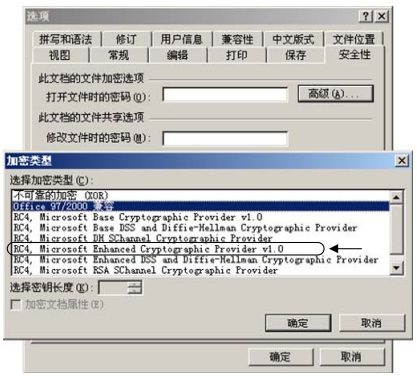

图1 Microsoft Word 2003中的文档安全性设置

因为在Office97~2003的实现中，“修改密码”只是个字段，所以在能打开Office文档的情况下，许多软件都可以通过重置该字段(无需“修改密码”)的方法解除Office文档的只读限制。

对于“打开密码”而言，Office真正实现RC4加密，该密码机制的安全性则完全依赖于RC4流加密的安全性。因为

作者简介：何克晶(1983－)，男，讲师、博士，主研方向：科学计算，并行计算，信息安全

收稿日期：2009-09-02 **E-mail：**kejinghe@ieee.org

“打开密码”不能简单破解，所以在一定程度上可以保护用户文档的安全和隐私。

Microsoft Office 系列是目前使用最为广泛的文档软件，所以，本文对 RC4 流加密和 Microsoft Office 密码机制进行研究，评价该加密机制的安全性，并给出加强 Office 文档安全性的建议。

本文的所有研究只利用完全公开的论文、程序和文档(如微软官方文档[7]、OpenOffice、wvWare 等)，不包含任何的反向工程内容。

3 RC4 与Microsoft Office密码机制

RC4 是种密钥长度可变的流加密算法，支持 40 bit~256 bit 的密钥长度。RC4 的核心部分的 S-box 的长度可变，但一般为 256 Byte。因为 RC4 加密的速度非常快(为 DES 的 10 倍左右)，实现简单，并且具有较高的安全性，所以得到广泛应用。

RC4 算法包括初始化算法和伪随机子密钥生成 2 个部分。初始化算法的主要目的是利用密钥把 S-box 搅乱。具体算法如下：

```
for i = 0 to 255:
    S[i] = i
j = 0
for i = 0 to 255:
    j = (j + S[i] + key[i mod keylength]) mod 256
    swap(S[i], S[j])
```

在初始化完成后，可用该 S-box 生成伪随机子密钥流。在生成密钥流的过程中，S-box 也会随着被打乱变化。

```
i = 0
j = 0
while GeneratingOutput:
    i = (i + 1) mod 256
    j = (j + S[i]) mod 256
    swap(S[i], S[j])
    output S[(S[i] + S[j]) mod 256] ^ input
```

尽管 Microsoft Office 的较新版本支持 128 bit 的 RC4 加密，但为了向下兼容 Office97，默认使用的都是 40 bit 的“Office 97/2000 兼容”RC4 加密。

Microsoft Office97~2003 系列使用微软复合文档(Windows Compound Binary File Format)格式[7-9]进行文件内容存储。微软复合文档是种结构化的二进制文件格式。在这种文件格式中，最初的 512 Byte 为文件头，定义整个文档内容的存储方式，而整个微软复合文档由若干流(Stream)组成。

在一个 Word(.doc)文档中，典型的流包括 Table 流(Table Stream)、文档流(Main Stream，也叫 WordDocument Stream)、数据流(Data Stream)和属性流(Summary Info Stream)等[7]。

若使用加密格式存储微软复合文档，属性流默认是不会加密的，而只有其他包含了文档数据的流才会被加密。其中，需要注意的是 WordDocument 流。

根据微软的官方文档[7]，在 Word Document 流的最前端包含了一个名为 FIB(File Information Block)的数据结构，这种结构几乎包含了正常打开微软复合文档所需要的所有重要信息。

使用加密格式存储时，FIB 的一些数据(如版本号，是否加密的状态位等)是明文存储的，而另外一些数据(如正文大小，尾注大小等)则是加密存储的。根据 wvWare 的文档，Word 文档的加密机制如图 2 所示。

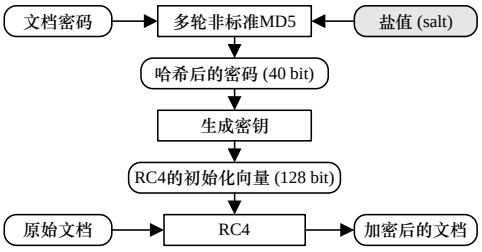

图 2 微软 Word 文档的加密机制

因为盐值(salt)的存在，使得每个文档即使密码相同，哈希后的密码(40 bit)也不一致，这就降低了预计算攻击的可能性。根据 wvWare 的文档，验证“打开密码”是否正确的机制如图 3 所示。

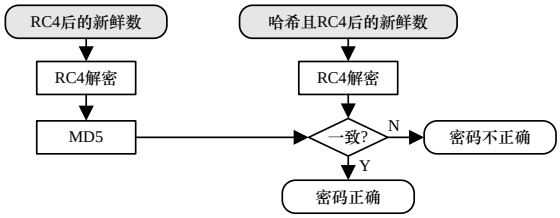

图 3 “打开密码”是否正确的验证机制

图 2 中的盐值以及图 3 中 2 个经不同方式加密过的新鲜数(灰色背景部分)都是 128 bit 的，都存储在 Table 流中，专供密码校验使用。Excel 和 Powerpoint 的加密机制类似。

4 密码机制的安全性

在不知道“打开密码”的情况下，要获得解密后的文件内容，一般有 2 种攻击方法：

(1)暴力破解法：暴力破解法是最简单、最常用的攻击方法。该方法通过结合图 2 的加密机制和图 3 的密码校验机制，通过不断猜测用户的原始密码来尝试对文件进行解密。

暴力破解法一般与密码字典结合使用。但因为 Office 文档可支持的原始密码最多达 15 个字符，且单次密码校验过程需要经过多轮比较慢的 MD5 操作，从而使得密码空间较大，破解速度较慢。

对于比较复杂的原始密码而言，只有通过分布式并行破解，或者使用专门的 FPGA 芯片才有可能在较短的时间内获得成功[6]。

(2)直接攻击哈希后的密码(40 bit)：若知道了图 2 中的 40 bit 哈希后的密码，在即使不知道原始“打开密码”的情况下，也可以解密整个文档。所以，该方法直接对哈希后的密码(40 bit)进行攻击。

这种攻击方法的好处在于绕过了多轮的非标准 MD5 操作，从而加快了单次攻击的速度。根据笔者在单个 Pentium 4 2.4 GHz 处理器上面的测试，从生成 40 bit 哈希后的密码到密码校验完成平均用时约 5 μs，即在这样单个处理器上穷举完整个 2^{40} 密码空间所需要的时间约为 64 天。

4.1 预计算攻击

上述攻击方法仍然利用了图 2 的加密机制和图 3 的密码校验机制。虽然绕过了图 2 中的加密盐值，但因为图 3 中 2 个新鲜数的存在，所以还是无法使用预计算攻击方法。而根据微软的官方格式文档，原始文档里的部分内容在加密前是已知的，即对于任意文档，该文档在固定位置的部分内容在加密前都是一样的。因为安全关系，本文并不说明可利用的已知内容包括哪些。这就可以利用图 2 的逻辑进行密码校

验，而无需使用图 3 的校验机制，如图 4 所示。

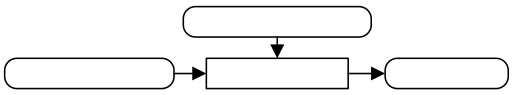

图 4　简化后的加密流程

从图 4 可以看到，在原始文档部分已知，而整个加密算法完全公开的情况下，哈希后的 40 bit 密码与加密后的文档之间存在着对应关系。

在图 4 中已经不存在任何盐值(salt)和新鲜数。这样就导致了更糟的局面——可以使用预计算攻击来破解该密码体系，进一步降低了复合文档的安全性。

4.2　Rainbow攻击

设 t 为执行图 4 所示单次攻击所需要的时间，若不进行预计算，攻击所需要的时间为 $2^{40}t$，这时需要的存储空间可忽略不计。若使用完整的预计算攻击，因为需要存储 40 bit 密码和加密后的文档之间的映射关系，这时的存储空间约为 $2^{40}\times40\times2$ bit，即约 10^4 GB=10 TB 的存储空间，而这时因为可以对映射关系进行预排序，再采用二分查找，所以可以实现非常快速的攻击。

Rainbow 攻击[10-11]是种在时间复杂度和空间复杂度之间取得一定平衡的攻击方法。在已知部分明文 $P0$ 和加密方法 S 的情况下，Rainbow 攻击还引入一种还原函数(reduction function)R。该还原函数的作用是根据密文 C 生成密钥 k。当然，真正的“还原”是不可能的。所以，这里的还原函数只是负责生成密钥 k，而不用管该密钥是否正确。很多时候在具体应用时，还原函数只是简单地复制或者哈希函数。在有了加密方法 S 和还原函数 R 之后，Rainbow 方法建立若干条密钥链：

$$k_i \xrightarrow{S_{k_i}(P_0)} C_i \xrightarrow{R(C_i)} k_{i+1} \tag{1}$$

Rainbow 方法的巧妙之处在于它只需要存储该密钥链的第 1 个密钥和最后 1 个密钥，这样就节省了大量的存储空间。最简单的情况下，对于一个密钥空间为 N 的问题，采用 Rainbow 方法，可以使用 $N^{2/3}$ 的存储空间，在 $N^{2/3}$ 的时间内进行破解。

4.3　验证实验

为了验证 Microsoft Office 文档密码保护的安全性，在本实验中，以 Word 为例，通过结合 Office 加密机制中的缺陷与 Rainbow 攻击解密 Word 文档。所有的实现均来自于 wvWare, libssl 和 libcrypto。所用的 Rainbow 参数如表 1 所示。

表 1　Rainbow 破解所用参数

参数	值	参数	值
密钥链长度	6 000	每个表中的密钥链数	3×10^8
预计算表数	6	存储空间	16.8 GB

在表 1 中，Pentium 4 2.4 GHz 处理器上每秒可以进行约 3×10^5 次从 k_i~k_{i+1} 的建链操作。整个预计算 Rainbow 表的时间约为 $(6\ 000\times3\times10^8\times6)/(3\times10^5)\approx3.6\times10^7$ s≈417 天。在本实验中，该工作被并行分布在 30 个处理器上进行，实际预计算 Rainbow 表的时间约 15 天。预计算任务只需执行一次，完成之后可供多次解密使用。

在预计算完成之后，每次使用单个处理器实际解密的时间平均只需约 1.5 min。根据 Rainbow 攻击的原理，与表 1 配置相对应的解密成功率可达 99.9%以上。本实验随机创建了 10 个 Word 文档，均成功快速解密。

4.4　Office文档安全性的加强

Office97~2003 文档默认的密码保护是非常脆弱的，不能用其保护重要的敏感信息。通常的解决方法有：

(1)使用 WinZip 和 WinRAR 等压缩软件的密码保护功能加密 Office 文档；

(2)选择内置的更高强度的 128 bit 的“Microsoft Enhanced Cryptographic Provider v1.0”加密方法加密。

本文推荐采用方法(1)。因为微软复合文档格式决定了已知两者的存在，所以可采用预计算攻击方法。这样就大大降低了密码体系的安全性，使得表面上看 128 bit 的密码体系，实际的强度可能远远低于 128 bit。最著名的例子是二战中 Bletchley Park 的密码学家们使用已知明文的破解技术破解了德国的 Enigma 密码机。而随着超大规模集成电路的发展，在单块普通的 FPGA 芯片上已每秒可搜索 10^7 个 RC4 密钥，使用专用大规模硬件已能较容易破解 64 bit 密码。所以，若密码体系不能保证真正的 128 bit 安全，为了稳妥起见，最可靠的方法是采用 WinZip, WinRAR 等软件内置的较安全加密方法(均使用 128 bit/256 bit AES 加密)进行再次加密。可喜的是，Office2007 默认使用的也是 128 bit 的 AES 加密，所以，相对 Office97~2003 而言，安全性大大提高。

5　结束语

本文分析 RC4 流加密及其在微软 Office 中的应用。通过分析发现，其所使用的默认 RC4 密码长度只有 40 bit，并且可以利用部分已知内容，绕过加密盐值(salt)和新鲜数，使用预计算方法进行攻击。通过结合 Rainbow 破解方法，在时间复杂度和空间复杂度之间取得平衡，本文用实验证明了使用一台普通计算机，即可在 100 s 以内解除该“打开密码”而获得文件内容。

通过完全公开的文档，而不需要借助任何反向工程技术，即能快速地解除 Office 文档的密码保护功能。因此，可以得出结论：Office97~2003 文档的密码保护是非常脆弱的，不能用其保护重要的敏感信息。通常的解决方法为采用压缩软件进行二次加密或者选择 Office 内置的更高强度的 128 bit 密码加密。

参考文献

[1] Scott R, Itsik M, Shamir A. Weaknesses in the Key Scheduling Algorithm of RC4[C]//Proc. of the 8th Annual International Workshop on Selected Areas in Cryptography. [S. l.]: ACM Press, 2001.

[2] 李　琴, 曾凡平. RC4 密码的改进方法及其性能分析[J]. 计算机工程, 2008, 34(18): 181-183.

[3] Nikita B, Ian G, David W. Intercepting Mobile Communications: The Insecurity of 802.11[C]//Pro. of the 7th Annual International Conference on Mobile Computing and Networking. New York, USA: ACM Press, 2001.

[4] Wu Hongjun. The Misuse of RC4 in Microsoft Word and Excel[Z]. (2005-01-10). http://eprint.iacr.org/2005/007.pdf.

[5] 张丽丽, 张玉清. 基于分布式计算的 RC4 加密算法的暴力破解[J]. 计算机工程与科学, 2008, 30(7): 15-17.

[6] Kwok S. Effective Uses of FPGAs for Brute-force Attack on RC4 Ciphers[J]. IEEE Transactions on Very Large Scale Integration (VLSI)Systems, 2008, 16(8): 1096-1100.

(下转第 135 页)