计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(8):80−89 [doi: 10.15888/j.cnki.csa.007564]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 3DES-RC4 混合加密的即时通信系统①

刘　峰[1], 王丹丹[1,2], 于　波[1], 于　飞[1]

[1](中国科学院 沈阳计算技术研究所, 沈阳 110168)
[2](中国科学院大学, 北京 100049)
通讯作者: 王丹丹, E-mail: wangdd_ucas@126.com

摘　要: 即时通信系统由于其实时性等特点已经成为一种重要的交流方式, 能够提高工作效率、降低沟通成本, 在企业、学校、政府等组织中扮演的角色越来越重要. 然而即时通信在带来便利的同时, 其固有的一些安全弱点阻碍了它的进一步发展. 为了保证即时通信系统的安全性, 一些先进的安全加密算法用于通信系统来防止攻击和信息泄露. 然而这些算法在加密强度或加密速度等方面都有各自的缺陷, 在理解了这些加密算法的局限性之后, 本文提出了一种旨在利用和组合两种加密算法最佳功能并提供比其中任何一种具有更好的安全性、实时性的替代算法, 即 3DES-RC4 混合加密算法, 是一种具有 256 个字节密钥空间的算法, 算法复杂度相较于 3DES 算法由 $O(2^{168})$ 提高到 $O(2^{5100})$. 基于此算法设计了一款即时通信系统, 针对系统的加密解密功能进行了测试, 分析了提出的算法的性能和强度. 并和 3DES 算法进行了对比, 证明了本文提出的算法保留了 3DES 加密强度和 RC4 伪随机性的特征, 在加密强度和适应性等方面优于构成算法.

关键词: 3DES; RC4; 安全加密算法; 混合加密; 即时通信系统

引用格式: 刘峰,王丹丹,于波,于飞.基于 3DES-RC4 混合加密的即时通信系统.计算机系统应用,2020,29(8):80–89. http://www.c-s-a.org.cn/1003-3254/7564.html

Instant Messaging System Based on Hybrid 3DES and RC4 Algorithm

LIU Feng[1], WANG Dan-Dan[1,2], YU Bo[1], YU Fei[1]

[1](Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)
[2](University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Instant messaging system (IMS) has become an important communication method due to its real-time characteristics, which can improve work efficiency and reduce communication costs. It plays an increasingly important role in enterprises, schools, governments and other organizations. However, while instant communication brings convenience, its inherent security weaknesses hinder its further development. In order to ensure the security of the instant messaging system, some advanced security encryption algorithms are used in the communication system to prevent attacks and information leakage. However, these algorithms have their own shortcomings in terms of encryption strength or encryption speed. After understanding the limitations of these encryption algorithms, in this study, we proposed an alternative algorithm that aims to leverage and combine the best features of both these algorithms and provide much better security than either of them, namely 3DES-RC4 hybrid encryption algorithm, is an algorithm with a 256-byte key space. The complexity is increased from $O(2^{168})$ to $O(2^{5100})$ compared with 3DES algorithm. Based on this algorithm, an instant communication system is designed. The encryption and decryption functions of the system are tested, and the performance and strength of the proposed algorithm are analyzed. By comparing to the 3DES algorithm, it is proved that the algorithm proposed in this paper retains the characteristics of 3DES encryption strength and RC4 pseudo-randomness, and is

① 收稿时间: 2020-01-13; 修改时间: 2020-02-08; 采用时间: 2020-03-11; csa 在线出版时间: 2020-07-29

superior to the constituent algorithms in terms of encryption strength and adaptability.

Key words: 3DES; RC4; security encryption algorithms; hybrid encryption; Instant Messaging System (IMS)

即时消息是一种两个或者两个以上的人基于键入文本进行实时通信的形式. 随着计算机网络的发展, 即时通信已经由最初的计算机专家快速交换重要信息的工具演变成全球最常用的通信交流机制之一[1]. 其功能也由最初的文本消息传递扩展到语音、视频通话以及图片视频等文件的实时传递. 即时消息在移动商务、移动银行、行政用途和日常生活等方面发挥着越来越重要的作用[2,3].

尽管即时通信技术会带来很多便利, 但同时也会引入很多风险和责任. 用户倾向于使用即时通信服务来传输所有类型的消息, 包括信用卡和银行账号等信息[4]. 攻击者将即时通信服务视为窃取信息的丰富来源, 监听是捕获通过网络传递即时通信消息最常用的方法[5]. 在公共即时消息传递系统中, 消息从客户端传送到服务器, 再传递到第二个客户端. 窃听者可能会沿着其 Internet 路径或网络内的任何地方看到此数据, 信息随时可能会传给其他人[6]. 因此对即时通信系统的消息传递进行安全加密, 保证消息的完整性和安全性是非常必要的.

安全加密消息传递是一种基于服务器的用于保护 Internet 上未授权访问的敏感数据的方法. 现代密码学涉及 4 个主要特征: (1) 机密性; (2) 完整性; (3) 不可拒绝性以及 (4) 身份验证[6]. 随着这些年来技术的发展, 用于即时消息加密的方法和技术也大大扩展, 对即时消息数据的加密强度也逐步增强.

计算机的出现使得我们保护消息发送而免受攻击的复杂性和速度呈指数级增长. 近年来, 诸如 AES、SHA 以及 3DES 等加密算法已经用于即时通信系统的消息加密以及解密[7−14]. 但是随着硬件的改进 (例如 GPU 的改进), 这些算法有可能被破坏. 由于此问题可能带来的威胁, 从长远来看, 还需要寻找替代方案来提高 Internet 上用户内容的安全性[2].

本文提出一种基于 3DES 和 RC4 的混合加密算法, 利用 3DES 加密强度和 RC4 算法的伪随机特征, 在保证加密速度的同时, 提高加密强度. 选择这两种算法的原因主要是: (1) 这两种算法同属对称加密算法, 加密速度快, 满足即时通信系统实时性的要求; (2) 模块化的 RC4 算法可以很轻易地替代 3DES 算法的默认密钥生成算法. RC4 算法的伪随机性质将改善原本由序列移位和压缩产生的可预见的 48 bit 的密钥的安全性.

1 加密算法介绍

针对信息传递的安全加密算法的研究有很多, 比如 3DES[11], AES[15], RSA[9,16] 等, 这些算法通常使用密钥来执行加密和解密的过程, 基于密钥的加密算法主要有两大类: (1) 对称加密算法; (2) 非对称加密算法 (公开密钥算法). 下文将介绍一些常用的安全加密算法.

1.1 DES 加密算法

DES (Data Encryption Standard) 加密算法是一种对称加密算法, 算法密钥长度为 64 位, 其中每 8 位中有一位作为奇偶校验位, 即实际密钥长度为 56 位. DES 算法由 4 个部分组成: (1) 初始置换; (2) 16 轮函数迭代计算; (3) 密钥生成算法; (4) 逆置换. 如图 1 所示.

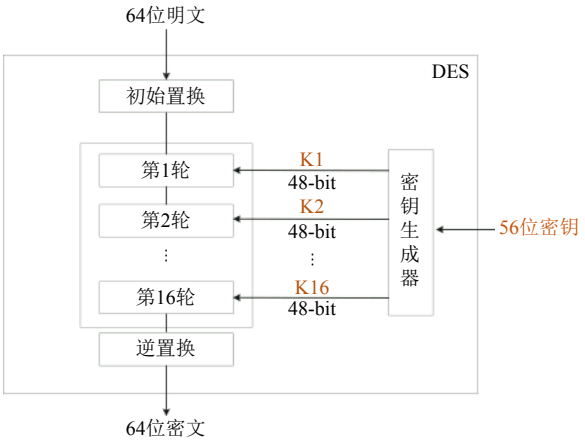


图 1 DES 加密算法框图

图 1 中初始置换和逆置换互为逆运算, 旨在将 16 轮函数迭代计算前后的结果按位进行重新组合, 以降低密文的统计特性并提高安全性. 16 轮函数迭代计算是 DES 算法的核心, 其利用 48 位的密钥对输入的低 32 位进行转换产生新的 32 位的输出, 作为数据块的高 32 位块馈入下一轮迭代计算函数. 密钥生成器产生一个 64 位的密钥作为输入, 并使用它按顺序生成 16 个不同的密钥用作函数迭代计算的输入密钥. 因此 DES

加密算法存在潜在弱点, 如果知道初始密钥, 一旦知道 P 盒置换, 就可以预测出 DES 算法的所有后续密钥.

1.2 3DES 加密算法

DES 算法的缺点之一就是其密钥空间很小, 容易被暴力破解. 因此 3DES 算法以 DES 算法作为基本模块, 以组合分组的方式设计出加密算法, 对数据块进行 3 次 DES 加密来扩大密钥空间. 其加密解密过程分别是对明文/密文数据进行 3 次 DES 加密或者解密, 每一次的密钥不同, 如图 2 所示. 3DES 加密算法有两个重要的参数, 即密钥数量和操作顺序. 3DES 可表示为: DES(k3;DES(k2;DES(k1;M))), 其中 M 是待加密的消息块, k1,k2,k3 分别为 3 个 DES 模块的密钥. 若 3 个密钥互不相同, 则相当于用一长度为 168 位的密钥进行加密. 若数据对安全性要求不高, 第一个密钥可以等于第 3 个密钥, 密钥的有效长度为 112 位[17]. 但由于其对特定明文和已知明文攻击的敏感性, 因此认为它只有 80 位的有限安全性[18,19]. 近些年来随着硬件计算速度的提升, 穷尽搜索 3DES 算法的密钥空间成为了可能.

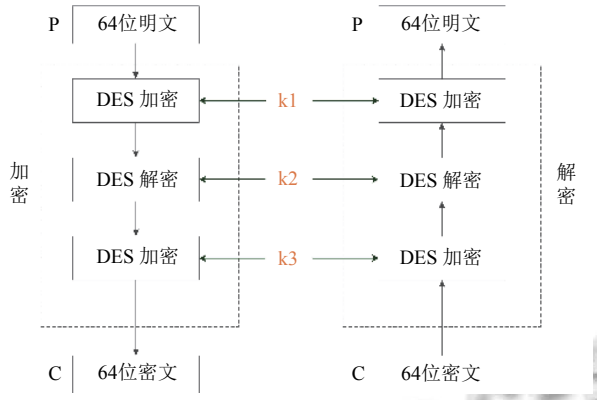


图 2 3DES 加密算法框图

1.3 RC4 加密算法

RC 加密算法是由 Rivest R 于 1987 年首次提出, 是一种对称加密算法. RC4 加密算法与 DES 算法不同, 它不是对明文进行分组处理, 而是以字节流的方式对明文中的每一个字节进行加密, 相同地, 解密的时候也是针对密文中的每一个字节进行解密. 该算法利用不规则置换[20], 其密钥长度可变, 在 8 位至 2048 位之间, 密钥流完全独立于明文[21,22]. RC4 加密算法强度很高, 至今尚未找对针对 128 位密钥长度的 RC4 算法的攻击方法.

RC4 加密算法流程如图 3. 状态向量 S 长度为 256 字节, 用于保持所有 8 位数的排列. 向量由长度为 1 到 256 个字节的密钥的排列初始化, 算法运行的任何时候, S 都包括 256 个 8 位的排列组合, 仅有值的位置发生了变换. 为了计算初始排列, 使用长度为 256 个字节的临时向量保存初始密钥, 如果密钥长度为 256 个字节, 则初始向量 T 就是初始密钥, 否则为初始密钥的重复排列. 密钥和临时向量 T 仅用于初始状态, 加密和解密状态仅和状态向量 S 相关. 密钥调度算法根据临时向量 T 初始化状态向量 S, 再通过伪随机数生成算法生成密钥流, 其长度和明文的长度是对应的. 密钥流和明文按字节进行异或就得到了密文.

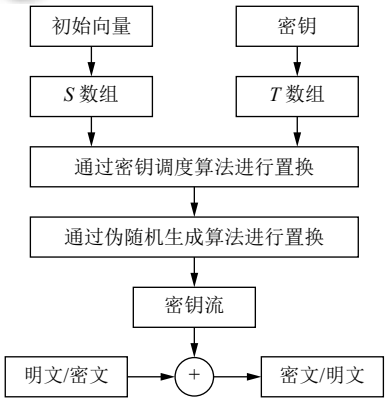


图 3 RC4 加密算法

2 3DES-RC4 加密算法

2.1 提出的 3DES-RC4 算法介绍

从上文对 DES 算法密钥如何生成的描述中可以看出, 其存在一个主要的缺陷. 如果知道了作为密钥生成算法的初始 64 位密钥, 则其后续的 16 轮迭代函数的密钥也能顺序破解. 另外, 如果知道了某一轮迭代的密钥, 也可以破解其他回合的密钥. 对于 RC4 加密算法, 即使知道了初始密钥, 其生成的密钥仍然是伪随机的, 其优势在于加密速度快, 安全性高, 可以抵御密码分析攻击.

基于以上分析, 本文提出了一种基于 3DES 和 RC4 的混合加密算法, 算法的框架类似于 3DES 算法, 一个主要的不同在于每一个 DES 算法中 16 轮迭代函数的密钥均由 RC4 加密算法提供. 利用 RC4 加密算法的伪随机数密钥生成机制, 以前一轮迭代函数的密钥或者初始密钥作为种子, 可以产生所有的密钥. 算法的加密过程如下:

$$K_i = RC4(key_i) \text{ xor } IV_i \quad (1)$$

$$K_{i,j} = PGRA(KSA(K_{i,j-1}), length) \quad (2)$$

其中, $K_i, 1 \le i \le 3$ 表示用于第i个 DES 过程的密钥子集和, $K_{i,j}, 1 \le i \le 3, 1 \le j \le 16$ 表示第i个 DES 过程的密钥子集和中第j轮迭代函数的子密钥.

算法. 3DES-RC4 算法

(1) 利用 RC4 伪随机数生成器产生 16 轮迭代函数的子密钥. 第一轮迭代函数的子密钥由初始密钥K_1作为种子, 其余轮迭代函数的子密钥均由前一轮迭代函数的密钥作为 RC4 加密算法的输入产生;

(2) 将产生的 16 个子密钥与初始密钥集 (IV) 进行异或操作产生新的 16 个 48 位的密钥, 作为 DES 算法中的每一轮迭代函数的输入;

(3) 通过上述两个规则, 中间密文经过解密和加密回合, 生成最终密文;

(4) 解密过程类似, 只不过密钥以相反的顺序提供给每一轮不同的 DES 过程.

3DES-RC4 加密算法的框架如图 4 所示. 图 5 表示了每轮 DES 如何通过 RC4 加密算法密钥生成函数获取密钥, 通过在发送方和接收方之间共享 3 个密钥 *key*1、*key*2 和 *key*3, 可以生成相同的伪随机密钥序列, 因此可以分别用于加密和解密过程. 这种方式可以确保不能利用任何中间回合密钥的信息获取原始密钥, 可以大大提高安全性能.

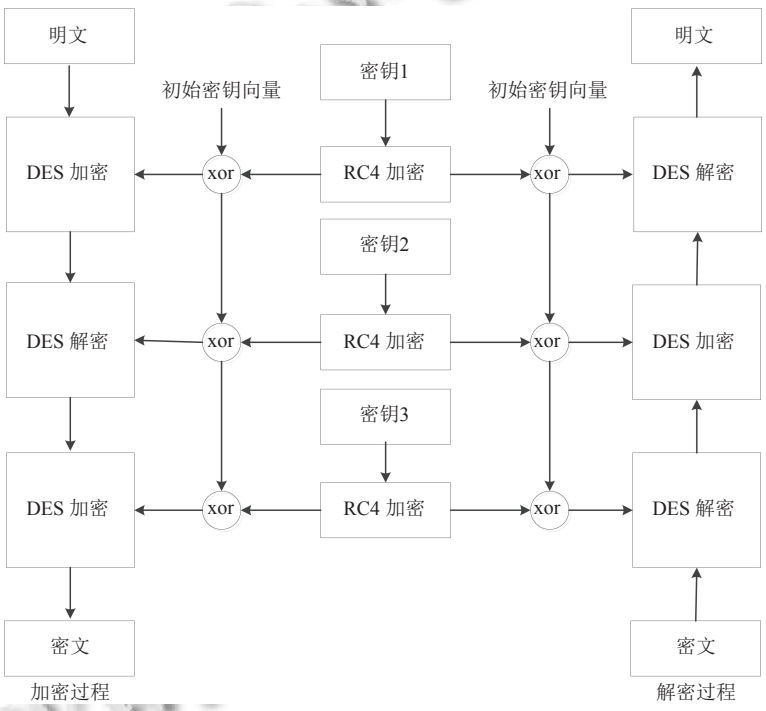


图 4 3DES-RC4 加密算法框架

2.2 3DES-RC4 算法复杂度分析

3DES-RC4 算法框架同 3DES 算法类似, 它们都需要额外的一个安全通道共享 3 个密钥, 而且都易于在硬件上实现. 与 3DES 算法不同的是: (1) 3DES 算法密钥安全性低, 而 3DES-RC4 加密算法生成的密钥是伪随机的, 安全性较高; (2) 3DES 算法使用迭代回溯方法获得密钥, 而 3DES-RC4 算法则是利用 RC4 算法生成密钥; (3) 3DES 算法中给每一个 DES 迭代函数的密钥是相互独立的, 而在 3DES-RC4 算法中, 本轮的密钥依赖于前一轮的密钥.

对 3DES 加密算法来说, 如果 3 个密钥互不相同, 则其复杂度为 $O(2^{168})$. 而 RC4 加密算法属于流密码体制, 其内部状态的大小是衡量其复杂度的一个重要因素. RC4 的内部状态由 256 个字节的 S 盒 2 个 8 bit 的指针 i 和 j 组成.根据 S、i 和 j 不同的取值,其内部状态可能的取值有 $2^{16} \times (256!)$ 种[23], 具有 $\log_2 2^{16} \times (256!) \approx 1700$ bit 信息量, 因此 RC4 算法复杂度为 $O(2^{1700})$. 本文提出的 3DES-RC4 加密算法, 核心思路是利用 RC4 加密算法

替代 DES 算法的密钥生成模块, 来扩大密钥空间, 因此 3DES-RC4 加密算法的复杂度为$O((2^{1700})^3)=O(2^{5100})$.

因此, 本文改进提出的 3DES-RC4 加密算法加密强度远远优于 3DES 加密算法. 同时由于其采用的算法框架类似于 3DES 加密算法, 故加密速度不会劣于 3DES 算法, 具体实验测试结果见后文.

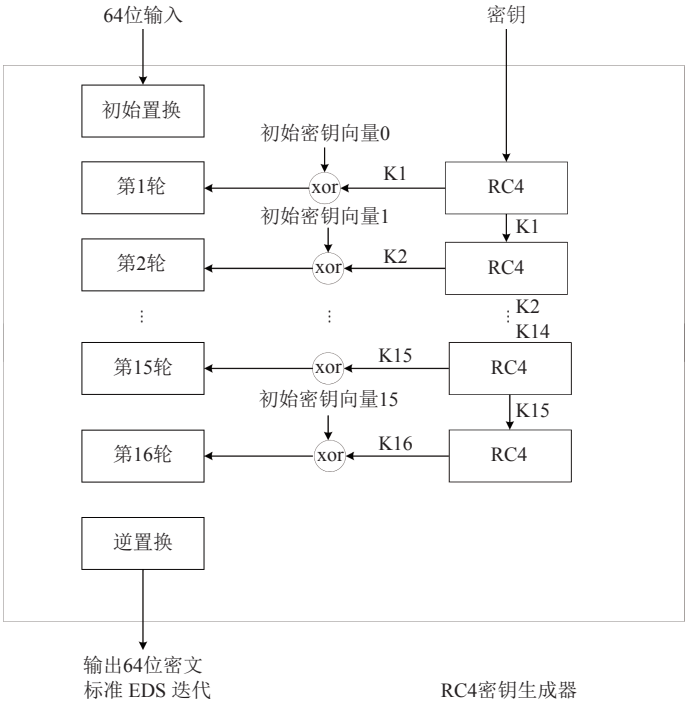


图 5 3DES-RC4 密钥生成

3 即时通信系统设计

3.1 即时通信系统框架

基于提出的 3DES-RC4 加密算法, 本文设计了一款即时通信系统. 通信系统提供登陆注册、通讯录、即时消息 (文字、语音、群聊) 等功能, 加密模块对文字、语音消息进行加密, 以提升整个系统的安全性. 即时通信系统密钥分配与会话过程时序图如图 6 所示. 登陆注册到服务器的客户端之间有一个共享密钥, 称为主密钥, 用户会话过程中, 由服务器通过主密钥为会话用户分配会话密钥. 客户端 A 和 B 登陆注册到服务器时, 由服务器对用户信息进行安全验证, 随后根据身份信息随机产生主密钥$K_A(K_{A1},K_{A2},K_{A3})$和$K_B(K_{B1},K_{B2},K_{B3})$, 并回应给用户 A 和 B. A 和 B 建立会话的过程如下, 首先由 A 向服务器发送会话请求及随机会话标识 N, 服务器对 A 的请求进行回应, 通过主密钥K_A对回应的消息进行加密, 因此只有 A 能成功解密这一消息. 回应的消息包括 A 需要的一次性会话密钥$K_S(K_{S1},K_{S2},K_{S3})$, 用于验证消息是否被篡改的 A 发送给服务器的请求信息以及 B 需要的由主密钥K_B加密的一次性会话密钥K_S, 用于对 A 身份进行验证的信息ID_A, 其中随机产生的K_S与会话标识 N 直接相关, 因此能保证会话密钥的定期更新. A 存储一次性会话密钥K_S, 并经服务器转发由主密钥K_B加密的会话密钥K_S和 A 的身份信息ID_A, B 收到密文后对其进行解密, 得到会话密钥K_S以及 A 的身份信息ID_A. 至此, 会话密钥K_S已经分配给用户 A 和用户 B. 如 A 要向 B 发送消息 P, 则 A 利用K_S对 P 加密得到密文 M, 经过服务器将 M 转发给用户 B. B 收到消息后, 利用会话密钥K_S对密文 M 解密得到消息 P, 并向 A 发送已读回执, 完成一次消息的发送.

3.2 即时消息加密模型

系统加密模块基于提出的 3DES-RC4 加密算法. 即时消息包括文字消息和语音消息, 针对不同的消息类型, 加密模型略有差异.

加密模型 1. 文字加密模型

(1) 发送方键入文字消息;

(2) 文字消息转化为字节序列;

(3) 利用 3DES-RC4 算法加密字节序列;

(4) 加密的字节序列转化位字符串;

(5) 将字符串传送给服务器;

(6) 接收方从服务器端接收加密字符串;

(7) 将加密字符串转化为字节序列;

(8) 对字节序列进行解密;

(9) 将解密的字节序列转换为同发送消息一样的字符串.

加密模型 2. 语音消息加密模型

(1) 发送方录入语音消息;

(2) 语音消息转换为字节序列;

(3) 利用 3DES-RC4 算法加密字节序列;

(4) 将加密字节序列存储到语音文件;

(5) 将语音文件放送到服务器;

(6) 接收方从服务器接收语音文件;

(7) 从接收的语音文件中提取出加密字节序列;

(8) 对加密字节序列进行解密;

(9) 将解密后的字节序列解析到文件输出流;

(10) 媒体播放器解析文件输出流并播放.

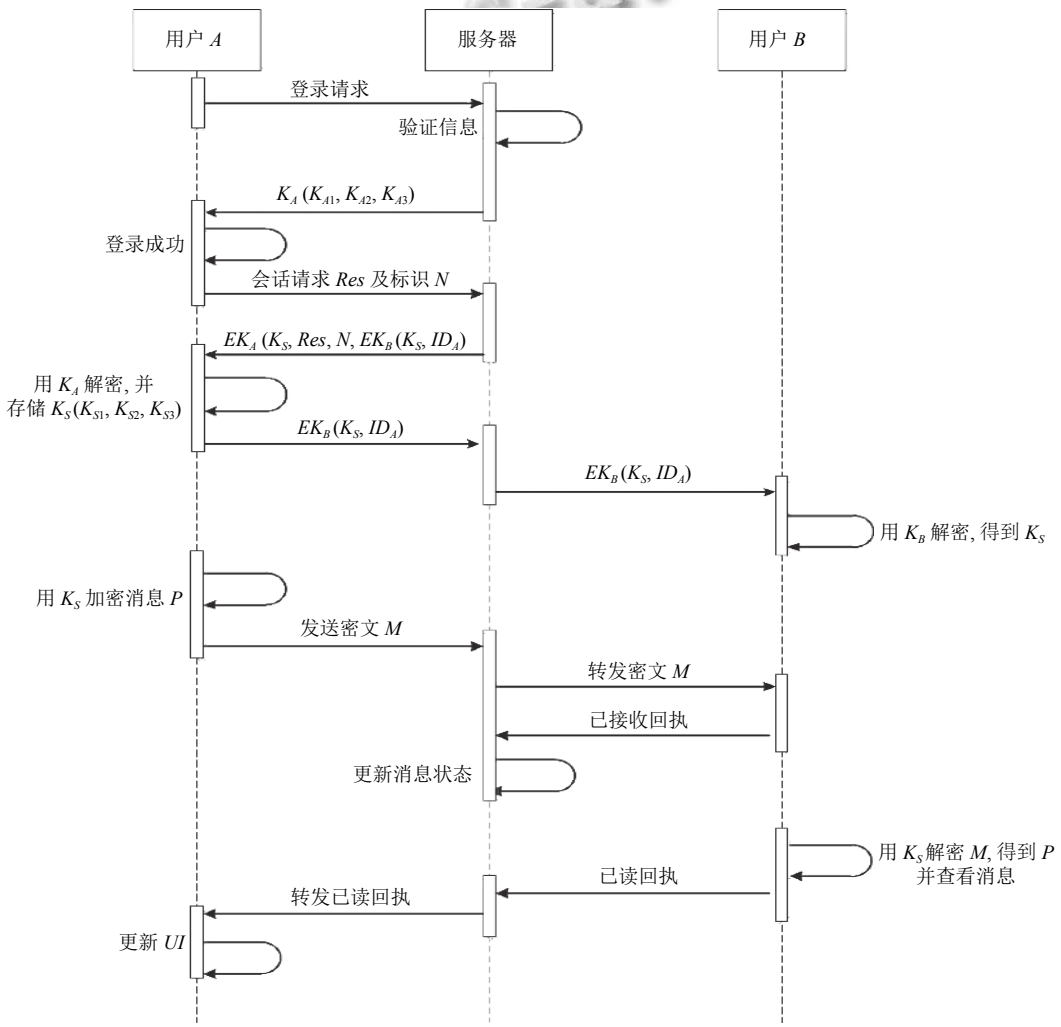


图 6 消息传递时序图

4 结果与讨论

为了对本文提出的 3DES-RC4 加密算法的功能和性能进行评估, 针对即时通信系统加密算法分别进行了加密和解密功能测试、加密性能测试、“雪崩效应”

验证及攻击模型验证等测试.

4.1 加密解密功能及性能测试

首先将以下内容作为输入明文: “Hello!世界”, 选取 3 个密钥分别为 *key*1: first key、*key*2:second key 和 *key*3: third key. 结合算法分析过程, 理论推导其密文, 为了方便对比, 明文、密文及中间过程密钥均采用 byte 数组表示.

原始明文: [72, 101, 108, 108, 111, 33, –28, –72, –106, –25, –107, –116]

密钥 1: [102, 105, 114, 115, 116, 32, 107, 101, 121]

密钥 2: [115, 101, 99, 111, 100, 32, 107, 101, 121]

密钥 3: [116, 104, 105, 114, 100, 32, 107, 101, 121]

初始向量 IV: [0, 0, 0, 0, 0, 0]

第一个 DES 子密钥集$K_{1,j}, 1 \le j \le 16$:

[[–60, –53, –36, 53, –109, –10],

[4, –58, –117, 76, 113, 32],

[15, –13, –31, 124, 81, –64],

[28, 44, –18, 90, –127, 106],

[55, –109, 77, 127, –68, 125],

[–64, 97, –123, –30, 85, 31],

[–107, –94, –50, 23, 55, –113],

[32, 35, 88, –31, 19, 53],

[114, –95, –90, 97, 124, 98],

[–48, –79, 83, –122, 26, –118],

[–45, 20, –22, –60, 11, –6],

[–103, 123, –16, –46, 45, 32],

[95, 109, –128, –60, –23, –80],

[64, –95, –30, –128, 75, –71],

[44, –47, 21, –79, –3, 79],

[–19, 6, –54, 67, –118, –8]]

明文经过第一次 DES 加密, 由于输入明文长度为 12, 而加密过程以 8 位为一个分块, 故在实际加密过程中, 需要将明文补 0 至 16 位, 密文为:

[68, 103, 58, –26, –121, 89, 57, –62, 56, –81, 67, 120, 115, 90, 3, 27]

第 2、3 次 DES 子密钥集产生方式类似, 故本文不再列出. 经过第 2 次 DES 过程之后的密文为:

[–6, –128, –87, 123, 20, –32, –94, –46, 11, 98, 6, 113, 6, 101, –45, –35]

第 3 次 DES 加密后的密文, 即明文经过 3DES-RC4 加密算法加密后的密文为:

[13, –34, 116, –68, –120, 9, 111, –5, 13, –5, –60, 21, –40, 30, –64, 74]

为了验证即时通信系统功能, 在测试中人为设定服务器回应的会话密钥与理论分析相同. 利用两个客户端进行通信, 客户端 *A* 向客户端 *B* 发送消息: “Hello!世界”, 日志如图 7 所示, 可以看出密文数据与理论计算相吻合, 证明了加密系统功能是正常的.

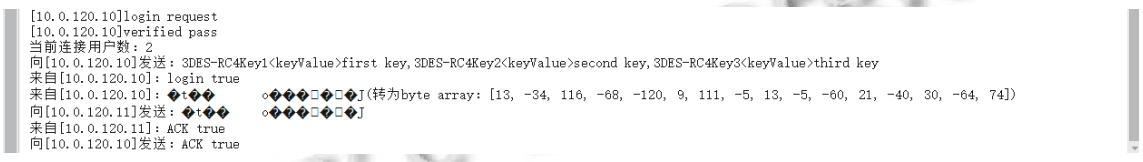


图 7 加密解密结果

为了进一步分析算法在移动平台的性能, 我们对基于此算法加密的移动即时通信 APP 进行了研究. 算法实现代码为 Java. 以 Redmi Note 7 手机为测试设备, 操作系统为 Android, CPU 为 4 个 2.2 GHz 和 4 个 1.8 GHz 的核心, 运行内存 4 GB, 存储空间 64 GB. 利用包含不同数量字符的消息对算法的性能进行了测试, 每个消息字符数量范围从 100 变化到 65 000. 为了对比, 同时测试了 3DES 算法的性能. 结果如图 8 所示, 可以看出 3DES-RC4 算法加密、解密时间与 3DES 算法近似相等. 在字符个数为 10 000 时, 加密时间约为 0.3 s, 表现出良好的时间性能. 由于此算法针对即时通信系统, 即时消息的长度一般不会超过 25 000, 其加密解密时间均小于 1 s. 故可以认为本文提出的 3DES-RC4 算法适用于即时通信系统.

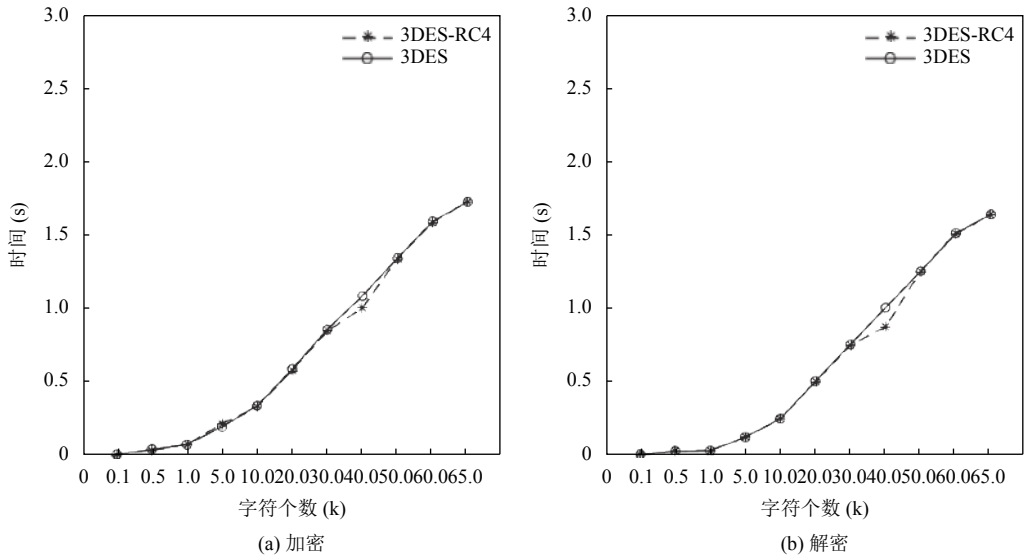
4.2 “雪崩效应”验证

为了对提出的 3DES-RC4 加密算法的加密强度进行进一步评估, 并和 3DES 加密算法进行对比, 分别测试分析了密钥和明文改变某一比特或某几比特时, 密文改变的比特数, 即两个算法的“雪崩效应”.

首先设定待加密明文为“Hello World!”, 初始密钥

为 *key*1:1、*key*2:2 以及 *key*3:3. 密钥 *key*2、*key*3 保持不变, 改变密钥 *key*1 的某一位或者某几位, 观察输出密文的变化情况, 结果如表 1 所示. 可以看到当密钥仅有很少的位数发生改变时就会引起输出密文很大的位数改变, 符合加密算法“雪崩效应”的准则.

在此基础上, 保持密钥为 *key*1:1、*key*2:2 和 *key*3:3, 设定初始待加密明文为 1. 改变输入明文的某一位或者某几位, 观察输出密文的变化情况, 结果如表 2 所示. 可以看出输入明文发生微小的改变时, 输出密文同样会产生剧烈的变化.



(a) 加密 (b) 解密

图 8 算法时间性能

表 1 密钥位数变化时密文位数变化情况

密钥1	密钥1变化位数	密文		密文变化位数	
		3DES-RC4	3DES	3DES-RC4	3DES
1	—	ü W, ./Oœ1§±&VJ¿ì+t˜ Ž š	• £§wèò´• Á R• ïæG Ç p•	—	
2	2	â"°ëQ ¼æ(èÐÒgš 2ð@Ûæ	Ú O5• å|nEÜ } Œ÷Ê¥ òŒ	101	96
3	1	Ú O5• å|nEÜ } Œ÷Ê¥ òŒ	v ©Ùd âV\],Ç³¯—¶ Áây×ô	114	94
8	3	0› ‘ ñ·Ê#Õ,D¤#¹g‘ Æ,L—í×	Ô{“ ¢Ÿ/ó'ú‹ (ì @Ö • Š ¢1æ×	92	89
31	3	ð-»¿í?¥ufµãQ¾@÷[IÄ!=#H	í pÙ¢ž ùßøƉ‰ ££%ýf¿• ‹ x	93	85

表 2 明文位数变化时密文位数变化情况

明文	明文变化位数	密文		密文变化位数	
		3DES-RC4	3DES	3DES-RC4	3DES
1	—	~óÃ½ Ÿ-Ó	³k!€ Ì÷#R	—	
2	2	ö|¥!Ow$.	Ð$pqŸQ¹Ž	34	32
3	1	ön¼ `_	MÇëèÉ	36	34
8	3	«ò «T• o³	e+àÍ¤ú=ø	31	28
31	3	(ã€ çWŠ 5	t Ã• R~½	29	27

根据实验结果, 3DES-RC4 加密算法和 3DES 加密算法均具有较好的“雪崩效应”效果. 改变密钥某一比特或改变明文某一比特时, 两个算法加密后的密文“改变位数”很大, 达到总数的一半左右. 同时对比可以发现, 在同样改变密钥或者明文的情况下, 3DES-RC4 加密算法的密文“改变位数”均稍大于 3DES 加密算法的密文“改变位数”. 因此 3DES-RC4 加密算法相较于 3DES 加密算法具有更好的“雪崩效应”, 加密强度更高.

4.3 攻击模型验证

加密算法的安全等级越高则意味着在实际使用过程中被攻击的难度越高. 为了对所提 3DES-RC4 加密算法的安全等级进行评估, 利用典型的攻击模型对算法进行了进一步的评估, 并和 3DES 算法进行对比.

穷举搜索攻击是最常见的攻击模型. 如前文分析, 3DES 复杂度为 $O(2^{168})$, 随着近年来硬件技术的发展, 在一定程度上可以穷举搜索攻击 3DES 算法. 而 3DES-RC4 加密算法的复杂度为 $O(2^{5100})$, 相较于 3DES 算法攻击难度大大增加.

已知明文攻击是攻击者知道明文密文对, 利用已知信息攻击密钥的一种攻击模式. 对于 3DES 加密算法, Ottawa 等已经证明在知道 2^{32} 明密文对时, 破解 3DES 加密算法的复杂度仅有 $O(2^{88})$, 这对于如今的硬件来说难度并不大[19]. 而 3DES-RC4 加密算法密钥生成模块 RC4 能够抵抗已知明文攻击[24], 故而已知明文攻击模型不能有效攻击提出的 3DES-RC4 算法.

选择明文攻击是另外一种攻击强度更高的攻击模式, 攻击者可以指定一定数量的明文, 让待攻击的加密算法进行加密, 得到相应的密文. Merkle-HeIlman 已经证明了 2-key 3DES 算法利用选择明文攻击时其时间和空间复杂度可以降至 $O(2^{56})$[18]. 由于 3DES 算法和本文提出的 3DES-RC4 加密算法均以 DES(16 轮) 算法加密为基础, 为了简单验证, 我们实现了 3DES 6 轮子过程的差分攻击算法模型, 分别对 3DES 和 3DES-RC4 算法子过程进行攻击实验, 攻击过程如图 9 所示. 结果如表 3 所示, 可以看出差分攻击可以攻击得到 3DES 的子密钥及初始密钥, 而对于 3DES-RC4 算法来说, 虽然可以攻击获得子密钥, 但是由于 RC4 算法对选择明文攻击免疫, 无法攻击得到初始密钥[24].

```
Encryption/decryption
difference  = 4008000004000000
plaintext   = 902E960DE90C6543
plaintext'  = D026960DED0C6543
key         = EAB9D6540E158054
difference  = 6B23AF2D27570DC5 (same in 0%)
ciphertext  = C11AF6AF6DA62B80
ciphertext' = AA3959824AF12645
ciphertext  = 63B65C638F5F3876 (final)
decrypted   = 902E960DE90C6543 (valid)
6-round attack
Valid key: [45, 56, 50, 26, 58, 35, 16, 40]
Key candidate: [-1, 56, -1, -1, 58, 35, 16, 40]
Key candidate: [45, 56, -1, 26, 58, 35, -1, -1]
```

图 9 差分攻击过程

表 3 差分攻击结果

加密算法	攻击实验次数	6轮子密钥能否破解	能否破解得到初始密钥	平均每次攻击时间(s)	密钥正确率
3DES	200	能	能	6.73	95.3%
3DES-RC4	200	能	否	—	—

5 结论与展望

本文提出了一种新的基于 3DES 和 RC4 的混合加密算法, 用于即时通信系统安全加密. 该算法使用 RC4 加密算法作为 3DES 加密算法的密钥生成器, 在保持加密解密速度不变的情况下有效地将密钥位宽从 64 位提高到了 2048 位, 算法复杂度由 $O(2^{168})$ 提升至 $O(2^{5100})$, 这使得该算法通过简单的密码分析很难破解. 除此之外, 算法使用了初始密钥向量, 确保一次密钥的信息不会损害其他密钥的保密性, 从而增强了算法的安全性. 以 3DES-RC4 算法为基础, 设计了一款即时通信系统, 对通信系统的加密解密功能、加密性能、加密强度进行了分析和研究, 结果证明了提出的 3DES-RC4 算法的可行性, 在加密强度和适应性等方面优于其构成算法. 接下来的研究工作是对此算法进行进一步的改进, 包括增加输出反馈机制等.

参考文献

1 Lewis J, Tucker B, Blake E. SoftBridge: An architecture for building IP-based bridges over the digital divide. Proceedings of the South African Telecommunications and Networking Application Conference. KwaZulu-Natal, South Africa. 2002.

2 Medani A, Gani A, Zakaria O, *et al*. Review of mobile short message service security issues and techniques towards the solution. Scientific Research and Essays, 2011, 6(6): 1147–1165.

3 Kabakuş AT, Kara R. Survey of instant messaging applications encryption methods. 2015. https://www.researchgate.net/publication/277867766_Survey_of_Instant_Messaging_Applications_Encryption_Methods.

4 刘赫德. 即时通信安全问题与措施. 中国设备工程, 2019, (7): 204–206.

5 袁庆军, 陆思奇, 韦忠兴, 等. 即时通信网络数据劫持分析研究. 信息网络安全, 2018, (11): 73–80. [doi: 10.3969/j.issn.1671-1122.2018.11.010]

6 Licari J. Best practices for instant messaging in business. Network Security, 2005, 2005(5): 4–7. [doi: 10.1016/S1353-4858(05)70233-6]

7 鲍海燕. 基于改进 AES 算法的网络数据安全加密方法研究. 信息技术与信息化, 2019, (9): 79–82. [doi: 10.3969/j.issn.1672-9528.2019.09.025]

8 李翔宇, 于景泽. DES 加密算法在保护文件传输中数据安全的应用. 信息技术与信息化, 2019, (3): 23–25.

9 李文胜. 基于 RSA 算法与对称加密算法的安全通信系统

的设计. 计算机安全, 2008, (6): 41–43. [doi: 10.3969/j.issn.1671-0428.2008.06.012]

10 Rahman MM, Akter T, Rahman A. Development of cryptography-based secure messaging system. Journal of Telecommunications System & Management, 2016, 5(3): 1000142.

11 Mandeep Singh Narula. Implementation of triple data encryption standard using verilog. IJARCSSE, 2014, 4(1): 1.

12 李峰. 基于 Android 平台即时通信应用的安全研究与实现 [硕士学位论文]. 上海: 上海交通大学, 2018. 1.

13 曾清扬. DES 加密算法的实现. 网络安全技术与应用, 2019, (7): 33–34. [doi: 10.3969/j.issn.1009-6833.2019.07.020]

14 张驰. 基于 DES 和 RSA 混合加密的即时通信系统的设计与实现 [硕士学位论文]. 厦门: 厦门大学, 2017.

15 Mandal AK, Parakash C, Tiwari A. Performance evaluation of cryptographic algorithms: DES and AES. Proceedings of 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science. Bhopal, India. 2012. 1–5.

16 Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, 21(2): 120–126. [doi: 10.1145/359340.359342]

17 Stewart JM, Chapple M, Gibson D. CISSP: Certified Information Systems Security Professional Study Guide. 6th ed. John Wiley & Sons, 2012.

18 Merkle RC, Hellman ME. On the security of multiple encryption. Communications of the ACM, 1981, 24(7): 465–467. [doi: 10.1145/358699.358718]

19 Van Oorschot PC, Wiener MJ. A known-plaintext attack on two-key triple encryption. In: Damgård IB, ed. Advances in Cryptology–EUROCRYPT'90. Berlin, Heidelberg: Springer, 1991. 318–325.

20 Stallings W. Cryptography and network security: Principles and practice, international edition: Principles and practice. International Journal of Engineering & Computer Science, 2012, 1(1): 121–136.

21 Kurt M, Duru N. Email encryption using RC4 algorithm. International Journal of Computer Applications, 2015, 130(14): 25–29. [doi: 10.5120/ijca2015907185]

22 王茂森. RC4 加密算法对无线网络安全技术的影响探究. 信息技术与信息化, 2018, (7): 184–185. [doi: 10.3969/j.issn.1672-9528.2018.07.057]

23 沈静. RC4 算法及其安全性分析 [硕士学位论文]. 广州: 广州大学, 2007.

24 Marwaha M, Bedi R, Singh A, *et al*. Comparative analysis of cryptographic algorithms. International Journal of Advanced Engineering Technology, 2013, 16–18.