同轴接入设备的系统升级机制设计

崔竞飞[1,2] 王劲林[2] 张 振*[3]

(1. 中国科学院大学 北京 100049; 2. 中国科学院声学研究所 国家网络新媒体工程技术研究中心, 北京 100190;
3. 北京大学 信息科学技术学院, 北京 100871)

摘 要: 高性能同轴网络(HINOC,High performance Network Over Coax)是面向下一代广播电视网的同轴电缆接入新技术,是实现模拟电视系统双向数字改造以及广播电视三网融合战略的关键技术之一. HINOC 系统中的嵌入式设备 HB(HINOC Bridge)与 HM(HINOC Modem)需要通过在线升级实现软件版本的不断更新. 针对目前一般的软件在线升级系统中普遍存在的整体加密传输效率低、硬件失效缺乏保护等问题,本升级系统设计在文件的传输和文件烧写的基本步骤上引入了文件差分编码解码、文件校验与加密解密以及异常保护机制. 实际测试结果显示,升级系统能够满足 HINOC 系统在效率、安全性和可靠性方面的要求,保证了嵌入式系统的信息安全以及嵌入式设备在异常情况下的稳定工作.

关 键 词: 高性能同轴网络(HINOC); 嵌入式设备; 系统升级; 文件编码; 加密解密; 异常保护

中图分类号: TN 915

文献标识码: A **文 章 编 号**: 1001-5965(2014)10-1355-06

Design of firmware upgrade algorithm for HINOC system

Cui Jingfei[1,2] Wang Jinlin[2] Zhang Zhen*[3]

(1. University of Chinese Academy of Sciences, Beijing 100049, China;
2. National Network New Media Engineering Research Center, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China;
3. School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China)

Abstract: High performance network over coax (HINOC) is a new technology of coaxial cable access, used for next generation broadcasting network. It can be used in triple play and the two-way cable TV network of digital and upgrade. The system is made up of HINOC bridge (HB) and HINOC modem (HM). In order to improve software function, it is necessary to update embedded software. File transmission and FLASH ROM program are basic processes in upgrades. In consideration of problems existed in widely used upgrades system such as low efficiency of encryption transmission and the lack protection for hardware failures, the system design introduced mechanisms such as coding and decoding of delta information, cryptography, file verification, and abnormal protection. According to the experimental results, the design meets requirements of HINOC in terms of efficiency, security and reliability. Information safety can be effectively protected, and the embedded devices can maintain basic function when abnormal problems happen.

Key words: high performance network over coax(HINOC); embedded device; system upgrades; coding and decoding; cryptography; abnormal protection

收稿日期: 2013-11-19; **网络出版时间**: 2014-03-13 17:37; **DOI**: 10.13700/j.bh.1001-5965.2013.0667
网络出版地址: www.cnki.net/kcms/doi/10.13700/j.bh.1001-5965.2013.0667.html
基金项目: 中国科学院重点部署项目(KGZD-EW-103-4); 国家 863 计划资助项目(2011AA01A106)
作者简介: 崔竞飞(1973 -),男,山东蓬莱人,教授级高级工程师,cuijingfei@abs.ac.cn; * **通讯作者**: zzbairen@gmail.com.

1 HINOC 系统简介

高性能同轴网络(HINOC, High performance Network Over Coax)系统是“三网融合”方案中由中国自主研发的有限同轴电缆宽带接入系统解决方案. HINOC 同轴电缆接入系统利用了现有的有线电视网络作为家庭用户接入网,主要由网桥(HB, HINOC Bridge)以及调制解调器(HM, HINOC Modem)组成. HB 通过光纤连接于光设备末端,作为与光设备唯一的传输接口,同时作为 HINOC 网络中心控制单元对网内的 HM 统一进行控制. HB 与 HM 通过同轴电缆连接,网管中心主机负责控制系统升级以及 HB 与 HM 的设备管理,升级服务器主要为远端设备提供升级文件[1-2], HINOC 系统架构示意图如图 1 所示[3].

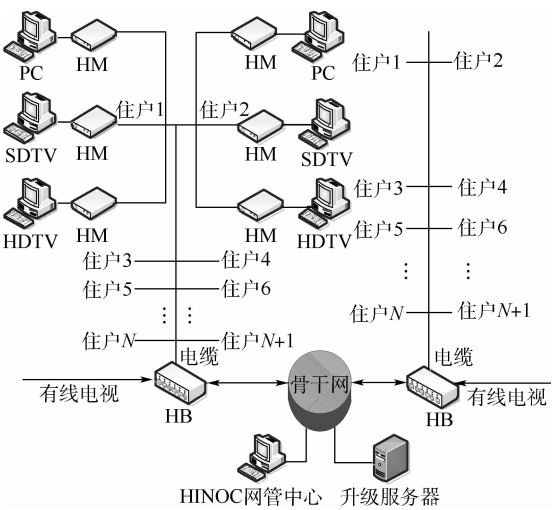


图 1 HINOC 架构示意图

Fig. 1 Structure of HINOC

系统升级设计是软件平台为了满足 HINOC 嵌入式设备软件系统升级的需要而设计的应用程序. HINOC 在投入用户使用后,随着软件版本的不断更新,需要采用远程在线升级来降低成本. 系统升级主要考虑升级的稳定性、文件传输以及更新过程的可靠性、升级系统通信的安全性等关键问题.

2 系统升级机制设计

基于嵌入式平台的升级系统传输的文件主要包括内核、文件系统、应用程序等. 传输方法常采用文件整体传输和差分文件传输[4]. 文件传输的过程中通信系统需要考虑文件保密、身份验证和数据完整. 数据完整校验常采用的方法包括 CRC 校验和单向散列函数校验,身份验证机制则包括对称加密与非对称加密[5]. 系统升级之后,需要进行系统和软件的重启,确保升级成功,为了保证远端设备在升级失败情况下基本功能的正常运行,需要引入异常保护机制[6].

2.1 系统升级流程

升级程序主要包括服务器端程序、客户端程序以及控制端程序. 服务器端程序运行在升级服务器上, HB 和 HM 运行客户端程序, HINOC 网管中心运行网络系统控制端. 升级服务器在启动前默认已经准备好升级所需要的文件,包括加密的密钥、需要传输的升级文件. 系统升级的序列图如图 2 所示,主要步骤如下:

1) 服务器端程序向网络控制端发送公钥,网管中心控制端将升级服务器地址以及解密公钥发送到升级客户端.

2) 客户端程序根据升级服务器地址发送升级请求,与服务器端建立链接.

3) 客户端接收升级文件以及加密的校验信息,通过校验信息和公钥检验文件传输是否有误.

4) 若文件传输正确,备份旧版本文件,若为应用程序,直接将文件存储至指定目录;若为内核文件系统或者 boot 脚本,则将文件烧写至 FLASH.

5) 通过系统和软件重启检验系统升级是否成功,若重启成功,则升级成功;若首次重启失败,启用异常保护机制,退出此次升级. 客户端同时向网管中心反馈升级状态信息.

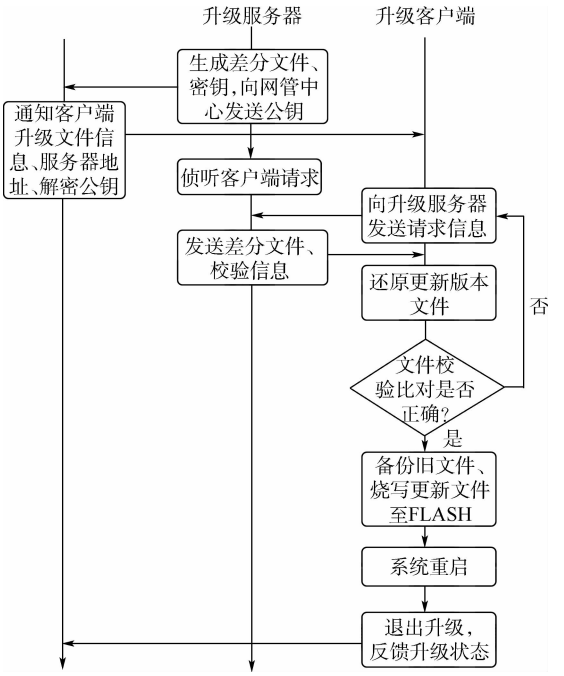


图 2 升级系统序列图

Fig. 2 Sequence diagram of upgrades

2.2　系统升级关键机制

升级系统的关键机制包括文件编码与解码、文件校验与加密解密以及异常保护机制.

2.2.1　文件编码与解码

在文件传输过程中,文件整体传输具有耗时长、效率低、安全性差等特点,容易造成信息泄露.在 HINOC 系统中,由于升级文件新旧版本文件之间改动较小,所以选择差分文件传输不仅可以节约传输时间,减少数据传输量,同时可以防止信息泄露[7-8].

最简单的差分算法是对不同版本的软件的字符块按字节逐个比较,而本系统中通过计算字符块的 Hash 值[9-10],采用 3 层 Hash 差分比较法生成差分文件,该算法具有计算速度快、碰撞概率小等特点. 主要步骤如下:

1) 将新版本文件以若干字节为一个字符块,分为若干块,分别计算块的弱 Hash 值,保存到一个 Hash 表中.

2) 取一个新版本字符块,与旧版本文件字符块进行匹配,首先比较两个字符块的前面几个字符. 如果不同,则移动一个字节,匹配旧版本文件下一个字符块;如果相同,则计算该字符块的弱 Hash 值,如果不同,则匹配下一字符块,如果相同,则进一步计算新旧两个字符块的 Hash 值,如果 3 层强 Hash 值相同则匹配成功. 记录旧版本标记、旧字符块在旧文件中的起始位置及块长度到一个结构体变量,并保存到差分文件中. 对于旧版本中连续匹配的字符块,只需改变结构体变量中的块长度,减少差分文件数据.

3) 如果到最后没有找到匹配的旧版本字符块,则匹配失败,记录新版本标记、新字符块在新版本文件中的起始位置及块长度到一个结构体变量,复制新版本字符块,并保存到差分文件中. 对新版本文件的各个块连续查找匹配块直到结束,便生成差分文件. 服务器端差分文件生成算法流程如图 3 所示.

在客户端,其还原算法如下:利用差分文件信息,逐个读取各个结构体数据,如果标记为旧版本数据,则利用块起始地址和块长度,从旧版本文件复制数据到还原文件;如果标记为新版本文件,则直接从差分文件复制数据到还原文件.

2.2.2　文件校验与加密

在数据校验的机制设计中,CRC 校验能很好地检测数据错误,但是因为 CRC 多项式是线性结构,在网络通信中,可以很容易被攻击者故意改变数据而维持 CRC 不变,因此 CRC 校验不适合网络数据校验. 单向散列函数校验能够对任意长度数据变化成固定长度输出数据,具有计算简单、数据不容易被故意更改等优点,常用的算法包括 MD5 和 SHA 等[9-12].

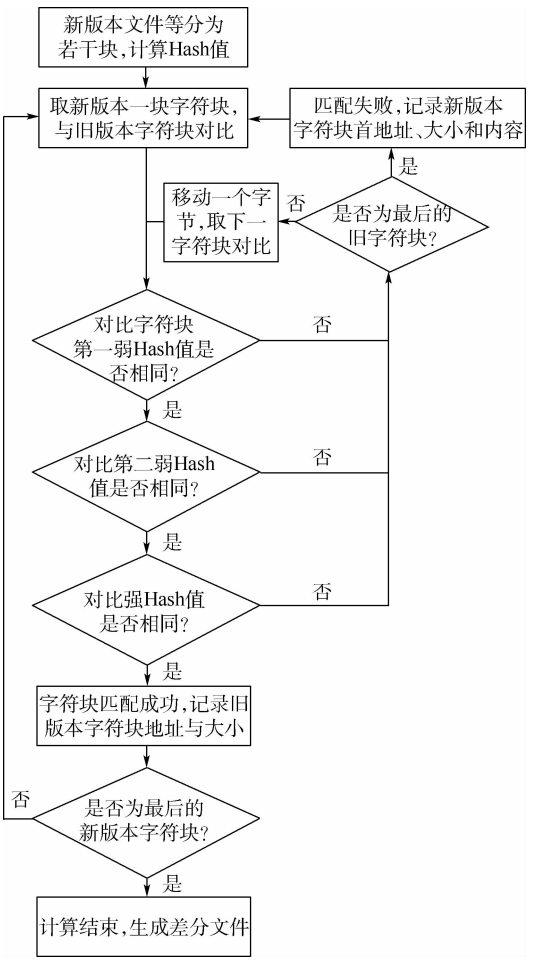


图 3　差分文件生成算法流程图

Fig. 3　Flow chart of delta file generating

在身份认证设计中,常见对称加密算法有 DES 和 AES 算法,其特点是同一密钥可同时用于加密和解密,收发双方使用同样的密钥,缺乏认证的功能,其安全性难以得到保障. 常用非对称加密主要有 RSA 和 DSA 算法,其特点是密钥分为公钥和私钥,私钥由收发其中一方保持,能够进行身份认证;而公钥可对外公开,方便密钥管理[13-15]. 综上,HINOC 系统采用单向散列函数(MD5 算法)校验结合非对称 RSA 加密验证机制.

升级服务器主要负责生成 RSA 私钥和公钥,并且将公钥通过网管中心发送至客户端. 升级服务器对更新版本文件进行计算产生 MD5 校验码,校验码经过 RSA 私钥加密,在差分文件传输完成后,再发送给升级客户端.

客户端收到加密的校验码后,利用 RSA 公钥解密出服务端的 MD5 码;在还原出升级文件的当

前版本之后,对其计算产生还原文件的 MD5 校验码. 前后两个 MD5 码进行比较,如果相等,则说明升级还原版本文件是完整的,并且发送方身份是可信任的;否则,说明文件传输出现问题,需要重新传输文件.

由于发送文件采用差分编码,所以即使被没有经过授权接受者获得,得到的文件信息也是无效的. 本升级系统结合了文件差分编码、文件校验和非对称加密等方式保障升级系统的安全性.

2.2.3 异常保护机制

系统异常主要指两方面:一是由于系统升级中途出现断电等硬件异常导致内核文件更新失败;二是由于系统内核文件区程序更新错误导致系统无法启动. 因此升级系统采用两种主要的保护方式来应对这两种异常情况.

1) 内核分区设计.

HINOC 系统采用两个内核文件映像分区方式,其中一个内核文件 FLASH 分区在内核驱动中设置为只读,在该分区存储出厂的内核与文件映像;另一个内核文件分区设置为可读写,该分区存储当前使用的内核与文件映像,当系统需要升级内核时,由于内核程序是运行在 RAM 中,所以升级程序可以对该 FLASH 分区进行擦除读写以更新内核,FLASH 分区设置如表 1 所示.

表 1 FLASH 分区设置

Table 1 Partition of flash device

分区	大小/MB	地址范围	读/写
Uboot 区	0.5	0x00000000-0x0007FFFF	读
出厂内核映像区	1.25	0x00080000-0x002BFFFF	读
出厂文件映像区	3	0x002C0000-0x004BFFFF	读
更新内核映像区	1.25	0x004C0000-0x005FFFFF	读/写
更新文件映像区	3	0x00600000-0x008FFFFF	读/写
启动环境变量区	0.25	0x00900000-0x0093FFFF	读/写
用户应用程序区	22.75	0x00940000-0x01FFFFFF	读/写

2) 内核重启机制.

HM 引入了自动重启机制解决系统异常问题,该机制通过系统环境变量判断启动内核版本,引入看门狗电路判断启动状态. 内核重启状态转移图如图 4 所示,主要步骤如下.

① 系统启动前,uboot 判断启动环境变量 flag1 的值:如果 flag1 = value1,boot 从出厂内核启动,flag1 值不变,将 flag2 置为 value1;如果 flag1 = value2,boot 从更新内核启动,将 flag1 置为value1, flag2 置为 value2.

② 系统启动后,如果启动成功,判断 flag2 的值:如果 flag2 = value1,可知系统运行在出厂内核状态,将 flag1 置为 value2,判断更新内核文件区是否错误,如果错误则取出备份内核文件映像,烧写到更新内核文件区;如果 flag2 = value2,可知系统运行在更新内核状态,将 flag1 置为 value2.

③ 系统启动后,如果启动失败:若更新内核启动失败,则看门狗电路自动重启设备,系统将从默认内核启动;若默认内核启动失败,则需手动烧写内核,由于默认内核是写保护,所以此类情况出现的可能性可以忽略.

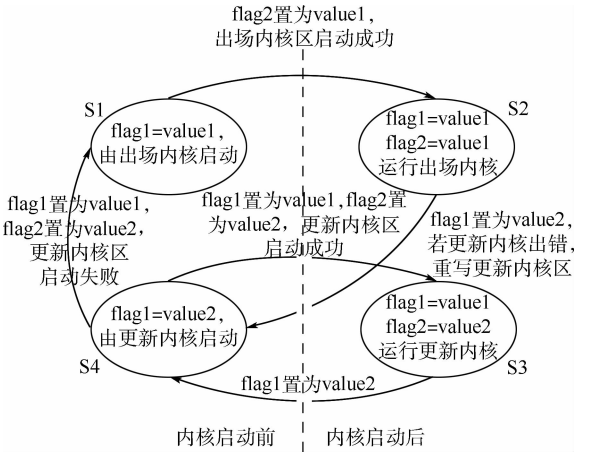


图 4 内核重启状态转移图

Fig. 4 State transition diagram of booting

3 系统测试

升级测试环境如图 5 所示,HM(左)通过同轴线与 HB(右)相连,HB 通过业务网口与升级服务器连接. HB 与 HM 硬件平台核心处理器是 XSCALE PXA270 嵌入式处理器,主频为 520 MHz,同时外接 32 MB 的 NOR FLASH 和 32 MB 的 SDRAM 存储器. 升级服务器和网管中心均使用 DELL OptiPlex960 主机,核心处理器为 Intel 酷睿 2 四核 Q9650 处理器.

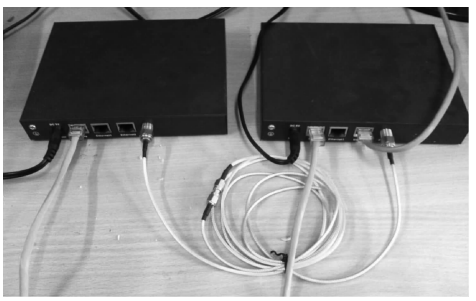


图 5 升级系统测试实物图

Fig. 5 Physical map of the upgrade system

3.1 差分编码功能测试

差分文件的生成在升级服务器中进行,差分文件主要为内核文件以及 HM 应用程序. 表 2 反映了内核和应用程序差分文件的生成和还原新版本耗时.

表 2　差分文件生成与还原

Table 2　Information of delta files

文件类型	文件/B			耗时/s	
	旧版本	新版本	差分	生成	还原
内核	806 432	806 432	132	5.31	0.29
应用程序	210 510	209 758	117 138	5.63	0.09

系统对不同文件进行差分文件的生成和还原,得到新版本文件正确,证明文件差分编码功能正确.

3.2　差分编码性能测试

图 6 为差分文件压缩率与比较字符块大小的变化曲线,随着字符块大小的增大,差分文件压缩率越大. 图 7 为差分文件生成时间与字符块大小的关系,随着字符块的不断增大,计算时间无明显变化趋势.

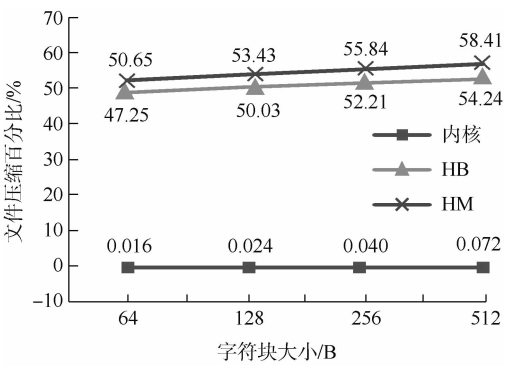


图 6　差分文件压缩率随字符块大小变化曲线

Fig. 6　Relationship between compression ratio and block size

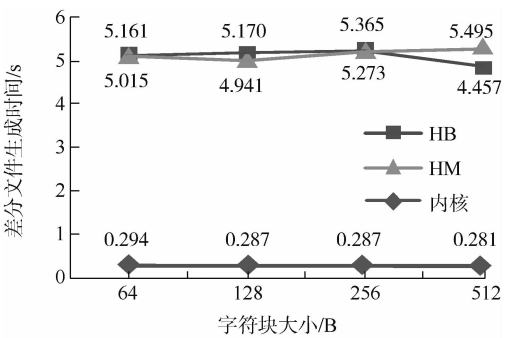


图 7　差分文件生成时间随字符块大小变化曲线

Fig. 7　Relationship between build time and block size

3.3　系统异常测试

系统异常测试主要是指嵌入式系统不能正常启动时,能自动判断错误并从出厂内核文件区启动. 由于 HM 第一版本系统硬件没有引入看门狗电路,因此重启时采用手动重启方式. 测试方法如下:破坏更新文件映像区,然后设置 boot 环境变量 flag1 为 value2,从更新内核文件区启动,测试结果如图 8 所示,启动 boot 时更新文件 CRC 校验出错.

```
   Image Name:   ramdisk
   Image Type:   ARM Linux RAMDisk Image (gzip compressed)
   Data Size:    3083473 Bytes =  2.9 MB
   Load Address: a0800000
   Entry Point:  a0800040
   Verifying Checksum ... Bad Data CRC
resetting ...
```

图 8　初次启动结果

Fig. 8　Result of first boot

手动重启后系统判断 flag1 和 flag2 环境变量分别为 value1 和 value2,判断内核文件启动失败,选择出厂内核文件区启动,完成异常条件下的成功启动,重启结果如图 9 所示.

```
Erased 1 sectors
## Booting kernel from Legacy Image at a0200000 ...
   Image Name:   kernel-2.4
   Image Type:   ARM Linux Kernel Image (uncompressed)
   Data Size:    806368 Bytes = 787.5 kB
   Load Address: a0200000
   Entry Point:  a0200040
   Verifying Checksum ... OK
   XIP Kernel Image ... OK
## Loading init Ramdisk from Legacy Image at a0800000 ...
   Image Name:   ramdisk
   Image Type:   ARM Linux RAMDisk Image (gzip compressed)
   Data Size:    3083473 Bytes =  2.9 MB
   Load Address: a0800000
   Entry Point:  a0800040
   Verifying Checksum ... OK
```

图 9　系统重启结果

Fig. 9　Result of restart

4　结 束 语

HINOC 系统升级程序的设计考虑了文件差分编码、文件信息加密、身份认证、异常保护等问题,应用范围广泛,可以适应多种嵌入式网络应用环境. 该升级系统相比普通升级系统具有以下优点:

1) 差分编码压缩传输文件大小,节省传输时间,保障信息安全;

2) 文件的校验和身份认证共同通过 MD5 码进行完成,相比其他升级系统,减少了信息交换时间,提高通信效率;

3) 异常保护机制保障了系统异常情况下设备的基本功能,提高了系统的可靠性,节省了入户维修的人力和经济成本.

由于 HM 硬件尚未引入看门狗电路,因此异常保护机制的自动重启功能仍需要进一步测试. 同时在实验环境下验证的基本功能,仍需要在实际应用环境中进行升级测试,才能进一步检验升级机制设计的完备性和健壮性.

参考文献 (References)

[1] GY/T 265—2012　NGB 宽带接主系统 HINOC 传输和媒质接入控制技术规范[S]
GY/T 265—2012　NGB broadband access system-technical specification of HINOC physical layer and media access control protocol[S](in Chinese)

[2] 欧阳锋,崔竞飞,赵玉萍,等. HINOC 同轴电缆接入系统技术方案[J]. 广播与电视技术,2011,38(10):34－38

Ouyang Feng, Cui Jingfei, Zhao Yuping, et al. Technology of high performance network over coax[J]. Radio & TV Broadcast Engineering, 2011, 38(10): 34 – 38(in Chinese)

[3] Cui W W, Duan X H, Yuan W, et al. Design and implementation of HINOC network management system based on SNMP[C]// Signal Processing, Communication and Computing (ICSPCC). Washington, D C: IEEE Computer Society, 2012: 527 – 532

[4] 陈璐,钟声. 网络文件传输方法和系统:中国,102420822[P]. 2012-04-28
Chen Lu, Zhong Sheng. Transfer method and system of network file: China, 102420822[P]. 2012-04-28(in Chinese)

[5] David A. Security enhanced firmware update procedures in embedded systems[D]. Linköping: Linköping University, 2008

[6] Andrew C. Secure firmware update: US, 2006143600[P]. 2006-07-06

[7] Ryozo K, Mariko K, Satoshi M, et al. A delta representation scheme for updating between versions of mobile phone software [J]. Electronics and Communications in Japan, 2007, 90(7): 26 – 37

[8] 王广辉. 嵌入式固件远程升级技术的研究与实现[D]. 成都:电子科技大学, 2011
Wang Guanghui. Design and implementation of remote upgrades based on embedded device[D]. Chengdu: University of Electronic Science and Technology of China, 2011(in Chinese)

[9] 张绍兰. 几类密码 Hash 函数的设计与安全性分析[D]. 北京:北京邮电大学, 2011
Zhang Shaolan. Design and security analysis on several cryptography hash functions[D]. Beijing: Beijing University of Posts and Telecommunications, 2011(in Chinese)

[10] 梁杰. MD5-Hash 函数的安全性分析[D]. 上海:上海交通大学, 2007
Liang Jie. Security analysis of Hash function MD5[D]. Shanghai: Shanghai Jiao Tong University, 2007(in Chinese)

[11] Putri Ratna A A, Dewi Purnamasari P, Shaugi A, et al. Analysis and comparison of MD5 and SHA-1 algorithm implementation in simple-O authentication based security system[C]//2013 International Conference on Quality in Research, QiR 2013-In Conjunction with ICCS 2013: The 2nd International Conference on Civic Space. Washington, D C: IEEE Computer Society, 2013: 99 – 104

[12] Järvinen K, Tommiska M, Skyttä J. Hardware implementation analysis of the MD5 hash algorithm[C]//Proceedings of the Annual Hawaii International Conference on System Sciences. Washington, D C: IEEE Computer Society, 2005: 298

[13] Ramesh A, Suruliandi A. Performance analysis of encryption algorithms for information security[C]//Proceedings of IEEE International Conference on Circuit, Power and Computing Technologies, ICCPCT 2013. Washington, D C: IEEE Computer Society, 2013: 840 – 844

[14] Stallings W, Brown L. Computer security principles and practice [M]. 2nd ed. Upper Saddle River, New Jersey: Prentice Hall, 2011: 71 – 101

[15] Wu X H, Zhou Y P. Analysis of data encryption algorithm based on WEB[C]//ICCET 2010-2010 International Conference on Computer Engineering and Technology, Proceedings. Piscataway, NJ: IEEE Computer Society, 2010, 7: 673 – 677