一种 Hash 特征隐藏的加盐信息摘要模型

祝彦斌,王春玲
(西安工程大学 计算机科学学院,陕西 西安 710048)

摘 要:介绍 MD5 和 SHA-1 等典型哈希函数现状,分析它们保持较高流行性的原因以及继续单独使用存在的问题。为解决现实信息摘要服务所面临的安全性和可用性问题,使用加盐技术、密码技术和若干基本信息摘要算法,设计一种加盐信息摘要模型。系统论述模型工作原理、设计和实现细节。在 OpenSSL 环境下实现一个原型,并且模拟它在网络通信中实用机制。结果显示,该模型可以隐藏基本 Hash 函数特征,产生更具随机性和抗碰撞性的摘要。最后,结合信息摘要在不同场景应用方式讨论该模型的盐值形式。

关键词:MD5; SHA-1; 哈希函数; 加盐; 信息摘要模型; OpenSSL
中图分类号:TP393;TP309.7 **文献标识码**:A **文章编号**:1673-629X(2013)03-0134-05
doi:10.3969/j.issn.1673-629X.2013.03.034

A Message-digest Model with Salt and Hidden Feature of Hash

ZHU Yan-bin,WANG Chun-ling
(College of Computer Science,Xi'an Polytechnic University,Xi'an 710048,China)

Abstract: Introduce the situation of typical Hash functions,such as MD5,SHA-1,etc.,and analyze the reason why they keep higher epidemic and the existing problem for continuing to use them alone. To solve the security and usability problem that message-digest service faces in reality,it designs a message-digest model with salting technique,encryption technique and some basic message-digest algorithms. And it systematically discusses the working principle and the design and implementation details of the model,realizes a prototype in OpenSSL environment,and simulates the practical mechanism in network communication. The results show that the model can hide characteristics of basic Hash functions and produce digests that have stronger randomness and resistance of collision. Finally,the salt form in the model is discussed combining with the way of message-digest applied in various scenes.

Key words: MD5; SHA-1; Hash function; salting; message-digest model; OpenSSL

0 引 言

典型哈希函数[1] MD5[2] 及 SHA-1[3] 作为国际电子签名和许多其它密码应用领域的关键技术[4],广泛应用于证券、金融等电子商务领域。2004 年中国山东大学王小云[5~7] 教授及其团队利用差分攻击法破解了 MD4、HAVAL-128、RIPEMD、SHA-0 等哈希函数。2009 年 11 月,文献[8] 提出了一种 MD5 密码并行攻击算法,并基于 GPU(Graphic Processing Unit,图形处理单元)设计和实现了 MD5 高速密码破解系统。当前许多站点和系统登录认证使用 MD5 保存密码。2010 年 11 月,一位德国安全爱好者花费 2 美元租用基于 GPU 的付费云计算资源暴力破解了 SHA-1 散列。2012 年 6 月,美国职业社交网站 LinkedIn 的 650 万用户密码被泄露,被泄露密码采用 SHA-1 算法加密。

典型哈希函数在实际使用中之所以保持较高流行度,一方面这些函数采用了直接设计[9] 的方法,具有良好的算法结构和性能,容易程序实现和应用;另一方面这些算法软件和硬件实现已经比较成熟,且容易获得。单独使用基本哈希函数,其特征很容易暴露,例如,MD5 摘要为 16Bytes,SHA-1 摘要为 20Bytes。账号口令等摘要一旦泄露,很容易遭受有针对性的攻击。文中结合加盐[10,11] 和密码方法提出一种信息摘要模型,随机选取算法,产生可隐藏 Hash 特征的、保持单向散列函数基本性质[12,13] 的摘要。

1 加盐信息摘要模型

1.1 概念简介

盐(salt),即随机数。盐值(salt value)来源包括:伪随机数产生器、用户指定、系统时间、内存随机区域等多种形式。加盐(salting),即引入随机数。加盐信息摘要模型(message-digest model with salt),就是在信

收稿日期:2012-07-07;修回日期:2012-10-10
基金项目:国家自然科学基金资助项目(60970140)
作者简介:祝彦斌(1985-),男,硕士研究生,研究方向为网络和信息安全;王春玲,副教授,主要研究方向为网络和信息安全。

息摘要模型中引入随机数,随机选取基本信息摘要算法(basic message-digest model) ,配合盐值完成信息摘要的计算和验证。

1.2　设计目标及原则

设计目标: 安全、透明地完成信息摘要计算和验证。

基本原则:

1) 哈希函数集合是可扩充的;

2) 哈希函数的选取是随机的;

3) 各摘要算法生成的摘要长度不尽相同,为隐藏算法特征,最终摘要长度(L_{end}) 应保持一致,并且要适应基本摘要长度(L_{basic}) ;

4) 由于 $L_{end} \geqslant L_{basic}$,为了保持最终摘要比特分布均匀,$L_{vsalt}$($L_{vsalt} = L_{end} - L_{basic}$) 部分至少与选取算法的摘要分布均匀程度相同。

1.3　模型工作原理

计算信息摘要时从预先设置的算法集合中随机选取一种算法完成基本摘要,使用密码算法对摘要算法标识进行加密,并作为摘要结果的一部分; 在验证信息摘要时,使用相同密码算法对摘要算法标识密文进行解密,提取对应的摘要算法对信息进行基本摘要计算,进而完成验证过程。模型摘要空间比现有算法摘要空间都要大,更容易抵抗穷举攻击和生日攻击,更容易减少碰撞机会。

模型工作过程中,盐值作为模型的关键信息,完成随机算法的初始化、作为填充数据等。摘要算法标识数据量很小,此时加密、解密对整个模型性能影响有限。由于盐值相对保密,企图通过暴力破解摘要算法标识密文而获取最终摘要信息的尝试是不现实的,因为一方面枚举每种摘要算法进行计算的成本与打算获得的信息价值相比较是得不偿失的,另一方面即使攻击者在枚举过程中恰好碰到了摘要算法标识明文,由于摘要算法标识的长度与盐值的长度规定不同,攻击者将无法碰到对应的盐值,不能正确完成后续的摘要步骤。

1.4　设计方案

加盐信息摘要模型的功能主要包括两个部分: ①计算信息摘要,如图 1 所示; ②验证信息摘要,如图 2 所示。

图示:

<<< : 对输入的字符串内容进行循环左移算法标识 index 位,例如输入字符串 strop 为: 0x 10325476 98badcfe efcdab89 67452301,则 strop<<<8 后对应的目标字符串 destop 为: 0x 32547698 badcfeef cdab8967 45230110;

⊞ : 长度相同的两个字符串按字节相加模 2^8 运算,例如字符串 strop1 为 0x 10325476 98badcfe efcdab89 67452301, strop2 为: 0x 32547698 badcfeef cdab8967 45230110,则 strop1 +strop2 后对应的目标字符串 destop 为: 0x 4286ca0e 5296daed bc7834f0 ac682411;

‖ : 长度任意的两个字符串进行连接操作,例如 strop1 ‖ strop2,则表示将 strop2 添加到 strop1 的尾部,即将 strop1 和 strop2 进行合并获得目标字符串;

⊕ : 对折异或操作,用来将长度为 length(偶数) 的字符串 strop 按照第 i 个字节和第 length - 1 - i 个字节进行异或,结果保存到第 i 个字节,i 为: 0,1,2,…,(length/2) -1。这样可以利用基本摘要算法的分布均匀性和扩散效应优化填充数据。

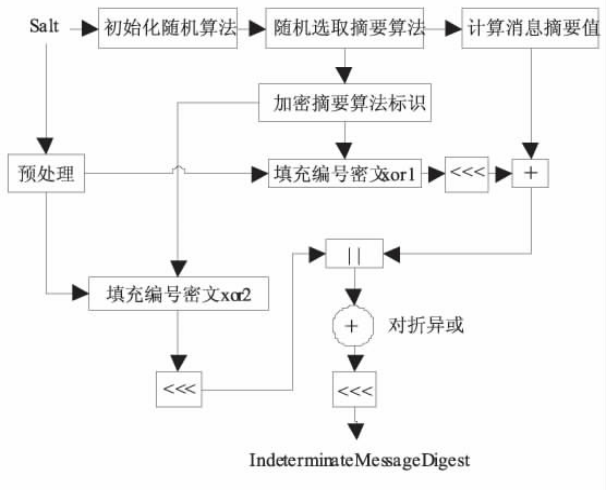

图 1　加盐信息摘要模型计算方案

填充编号密文 xor1:

将加密算法输出的密文 ciphertype 进行填充,填充的数据来自对 Salt Value 的预处理,填充后 xor1 的长度和当前随机选择的基本摘要算法摘要长度相同,例如当前选择的摘要算法如果是 MD5,则填充后 xor1 的长度为 16Bytes,即 128bits;

填充编号密文 xor2:

将加密算法输出的密文 ciphertype 进行填充,填充的数据来自对 Salt Value 的预处理,填充后 xor2 的长度为: L_{vsalt},即 $L_{end} - L_{basic}$。

预处理:

以 Salt 为数据源产生填充数据,如果填充数据长度小于或等于数据源长度,则填充数据直接来自数据源,如果填充数据长度大于数据源长度,则在重复使用数据源时需要对源数据进行必要的异或、循环左移等增量处理,以增强填充数据的随机性。

加盐信息摘要模型验证方案中的操作与模型计算方案中的基本操作相同,只是摘要算法编号密文来自存储或对方发送的数据。

模型在设计上采用模块化方法,在同一模块完成摘要计算和验证两种功能,这样模型可以相对独立地

引入到实际应用系统中，并且模块采用统一的接口向外提供服务，简化模型的使用。

2 模型原型实现及实用机制模拟

2.1 原型中基本算法概述

模型原型构建初期，使用 MD2、MD4、MD5、RIPEMD160、SHA 和 SHA-1 作为基本摘要算法集合中的元素，后续可扩充更多改进的和新的摘要算法元素，不断增强摘要的随机性。

构建过程中，使用对称加密算法 RC4 对信息摘要算法标识进行加密和解密，为信息摘要算法标识的保密提供基础保障。

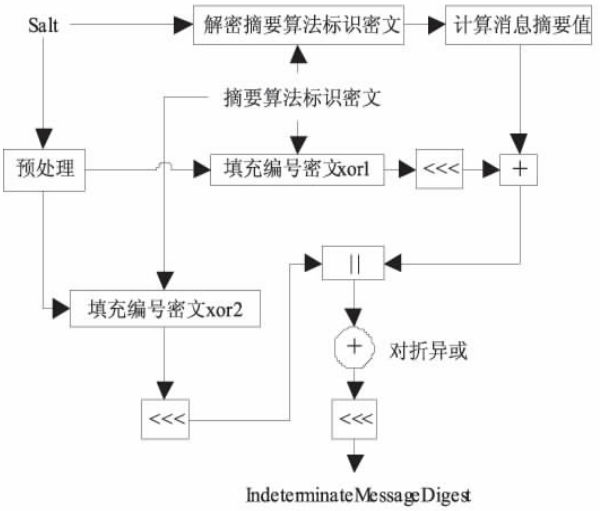

图 2 加盐信息摘要模型验证方案

RC4 是 Ronald Linn Rivest 在 1987 年为 RSA Security 公司设计的密钥大小可变的流密码。使用 RC4 主要是由于流密码与分组密码相比几乎总是更快，使用更少的代码。这一点也是原型使用 RC4 的主要原因。虽然 RC4 存在两个明文使用同一密钥进行流密码加密，将对应的两个密文进行异或得到原始明文的异或值的缺陷，但对于加盐信息摘要模型不存在此问题，因为在此模型中，明文(信息摘要算法的标识)选取是系统随机自动产生，人为将无法直接参与，所以选择明文攻击在此无法实施。

2.2 原型实现

在 Windows 平台配置 OpenSSL 环境实现该原型。配置和使用方法参见：http://www.openssl.org/。

在实现过程中需要注意以下几个方面：

1. 原型在实现时使用 OpenSSL 完成对随机数的产生以及基本摘要算法的实现，需要加载的库为 libeay32.lib，需要的程序模块有：crypto、rand、rc4、md2、md4、md5、ripemd 和 sha。

2. 定义原型摘要长度：

```
#define INDETERMINACY_DIGEST_LENGTH 70
```

3. 原型使用的基本摘要算法集合：#define TOTAL_DGST_ALGORITHM 6，即集合暂时包括 6 种基本摘要算法：MD2、MD4、MD5、RIPEMD160、SHA 和 SHA-1，具体定义为：

```
typedef enum BasicDigestAlgorithm{ MD2 = 1, MD4, MD5, RIPEMD160, SHA, SHA-1}
```

4. 定义原型摘要结构：

该结构包含摘要算法标识密文，用 8bits 表示算法标识 index(0 ~ 255) 的密文，以及计算的散列值 IndeterminateMessageDigest，用 70Bytes，即 560bits 表示，便于适应不同摘要算法产生的摘要长度。该结构作用非常重要，它用来传递原型的摘要计算结果和在验证时作为参数传递给原型进行计算比较。

5. 用 Salt Value 对 OpenSSL 随机数产生器进行初始化，具体使用函数为：

```
void RAND_seed( const void * buf, int num); int RAND_pseudo_bytes( unsigned char * buf, int num)
```

6. 随机数产生器产生的 buf 通过数据类型转换对 TOTAL_DGST_ALGORITHM 求模运算获得随机算法标识，然后严格按照模型设计方案实现程序。

7. 原型实现形式将以库的形式提供给使用者，使用者只要按照一般库文件进行加载并按照规定使用即可，导出服务接口函数为：

```
#ifdef __cplusplus
extern "C" {
#endif
/* 导出函数的名称* /
int fnIndeterminateDgstService(
/* 盐值* /
char * saltvalue,
/* 需要进行摘要的输入数据* /
char * message,
/* 可指定对文件进行摘要计算* /
char * filepath,
/* 输出原型摘要结构结果或验证输入的结构值* /
INDETERMINACY_DIGEST * retdgst,
/* 输出原型摘要长度* /
int * dgstlength,
/* 输出的验证结果* /
int * verified,
/* 验证时输出的当前摘要结构结果* /
INDETERMINACY_DIGEST * curdgst
);
#ifdef __cplusplus
}
```

#endif

2.3　原型实用机制模拟

为验证模型有效性，文中利用 Socket 网络通信技术模拟信息发送和接收交互过程中使用原型进行计算和验证摘要，整个流程如图 3 所示，实验结果分别如图 4 和图 5 所示。

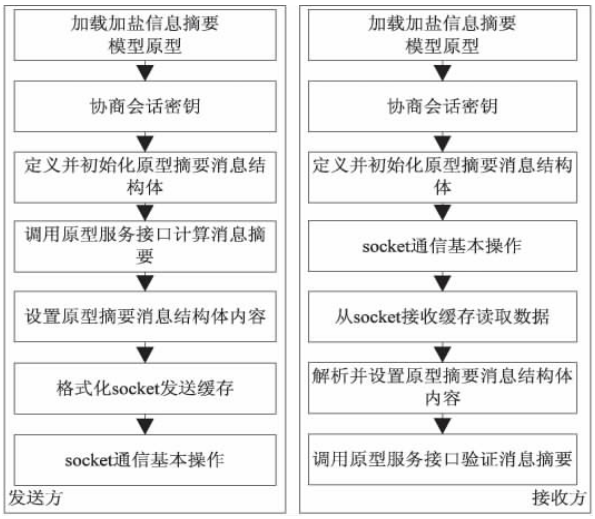

图 3　Socket 网络通信模拟原型实用机制

```
sender> Please input your key
wangcl_zhuyb_xpu_699
sender> Please input message
Hi, how are you? I send a test message to you
sender> 使用原型计算摘要值
sender> com1withcom2输出摘要值
0f:bd:85:a3:9c:23:e8:dd:05:3a:0f:6e:1b:85:8a:da:9f:5b:d3:14:7f:7f:c3:
1d:04:22:3c:b1:d4:aa:0e:e7:51:98:72:04:47:ed:87:52:9d:d8:5b:99:d8:db:
17:de:9a:1d:5e:58:97:ce:c3:94:5d:8c:33:71:0b:5d:8f:3a:ea:e5:a3:ad:58:ec
sender> successfully called sendto 16489
sender> Please input message
Hi, how are you? I send a test message to you
sender> 使用原型计算摘要值
sender> com1withcom2输出摘要值
0c:1a:f9:1f:04:63:5f:1f:a8:51:b1:6f:1a:d2:76:60:c1:d8:f7:a7:2f:8d:55:
cd:2b:0e:c5:c5:a8:be:fb:89:d3:d2:f5:a6:ee:f4:7e:3d:b7:76:16:e6:76:36:
c5:f7:a6:87:57:96:25:f2:20:44:e8:3f:1e:e4:d8:c8:62:a9:d3:5a:17:50:b1:26
sender> successfully called sendto 16489
sender> Please input message
```

图 4　Socket 发送方使用原型

实验结果分析:

对比图 4 和图 5 中数据显示: 两次通信的消息内容相同，通过模型原型计算，两次输出的摘要值不同，表明原型随机选取了不同的基本摘要算法进行摘要计算; 两次通信接收方使用原型进行消息的摘要验证结果相同，表明模型满足了散列函数的基本性质，即使用同一摘要算法对同一信息进行摘要计算的结果相同，还表明通信交互过程中发送方和接收方使用的盐值相同，即完成用户身份认证;

原型产生的摘要(按照十六进制格式化) ，即第一次摘要:

0f: bd: 85: a3: 9c: 23: e8: dd: 05: 3a: 0f: 6e: 1b: 85: 8a: da: 9f: 5b: d3: 14: 7f: 7f: c3: 1d: 04: 22: 3c: b1: d4: aa: 0e: e7: 51: 98: 72: 04: 47: ed: 87: 52: 9d: d8: 5b: 99: d8: db: 17: de: 9a: 1d: 5e: 58: 97: ce: c3: 94: 5d: 8c: 33: 71: 0b: 5d: 8f: 3a: ea: e5: a3: ad: 58: ec

```
receiver> Please input key
wangcl_zhuyb_xpu_699
receiver> Waiting for new message..
receiver> ################Received new message################
receiver> Hi, how are you? I send a test message to you
receiver> 使用原型验证摘要值
当前计算的消息的摘要值为:
0f:bd:85:a3:9c:23:e8:dd:05:3a:0f:6e:1b:85:8a:da:9f:5b:d3:14:7f:7f:c3:
1d:04:22:3c:b1:d4:aa:0e:e7:51:98:72:04:47:ed:87:52:9d:d8:5b:99:d8:db:
17:de:9a:1d:5e:58:97:ce:c3:94:5d:8c:33:71:0b:5d:8f:3a:ea:e5:a3:ad:58:ec
receiver> com2withcom1 接收的摘要值
0f:bd:85:a3:9c:23:e8:dd:05:3a:0f:6e:1b:85:8a:da:9f:5b:d3:14:7f:7f:c3:
1d:04:22:3c:b1:d4:aa:0e:e7:51:98:72:04:47:ed:87:52:9d:d8:5b:99:d8:db:
17:de:9a:1d:5e:58:97:ce:c3:94:5d:8c:33:71:0b:5d:8f:3a:ea:e5:a3:ad:58:ec
摘要值校验结果 相同。
receiver> Waiting for new message..
receiver> ################Received new message################
receiver> Hi, how are you? I send a test message to you
receiver> 使用原型验证摘要值
当前计算的消息的摘要值为:
0c:1a:f9:1f:04:63:5f:1f:a8:51:b1:6f:1a:d2:76:60:c1:d8:f7:a7:2f:8d:55:
cd:2b:0e:c5:c5:a8:be:fb:89:d3:d2:f5:a6:ee:f4:7e:3d:b7:76:16:e6:76:36:
c5:f7:a6:87:57:96:25:f2:20:44:e8:3f:1e:e4:d8:c8:62:a9:d3:5a:17:50:b1:26
receiver> com2withcom1 接收的摘要值
0c:1a:f9:1f:04:63:5f:1f:a8:51:b1:6f:1a:d2:76:60:c1:d8:f7:a7:2f:8d:55:
cd:2b:0e:c5:c5:a8:be:fb:89:d3:d2:f5:a6:ee:f4:7e:3d:b7:76:16:e6:76:36:
c5:f7:a6:87:57:96:25:f2:20:44:e8:3f:1e:e4:d8:c8:62:a9:d3:5a:17:50:b1:26
摘要值校验结果 相同。
receiver> Waiting for new message..
```

图 5　Socket 接收方使用原型

第二次摘要:

0c: 1a: f9: 1f: 04: 63: 5f: 1f: a8: 51: b1: 6f: 1a: d2: 76: 60: c1: d8: f7: a7: 2f: 8d: 55: cd: 2b: 0e: c5: c5: a8: be: fb: 89: d3: d2: f5: a6: ee: f4: 7e: 3d: b7: 76: 16: e6: 76: 36: c5: f7: a6: 87: 57: 96: 25: f2: 20: 44: e8: 3f: 1e: e4: d8: c8: 62: a9: d3: 5a: 17: 50: b1: 26

表明摘要长度和预期相同，即隐藏了基本算法特征，为 70Bytes，即 560bits，理论上抗强碰撞攻击的复杂度为 2^{280}，抗弱碰撞攻击的复杂度为 2^{560}，远远高于目前任意单一摘要算法同类复杂度。摘要从形式上也表现了分布均匀性，可大大降低摘要冲突机会。

3　模型应用场景

信息摘要应用场景不同，模型的盐值来源和形式也将发生相应变化，概括起来可分为以下 4 种场景:

(1) 网络通信场景: 通信双方预先协商密钥。如果使用对称加密算法，双方持有的共享密钥作为盐值，验证时模型还需要对方发送的摘要和算法标识密文; 如果使用非对称加密算法，模型计算和验证时都要将公钥作为盐值，计算时发送方使用私钥对算法标识加密，验证时接收方需要对方发送的摘要和算法标识密文，并使用公钥解密密文。

(2) 远程站点登录场景: 用户登录口令作为模型的盐值，模型输出的算法标识密文和摘要与用户账号

对应写入数据库，便于下次登录验证。

(3) 组织内(例如企业内部) 场景：模型的盐值可以是关于组织的多维特色信息。组织内各种具体应用场景要统一盐值，便于互相验证信息摘要。

(4) 本地应用场景：模型的盐值可以是系统管理员口令或管理员指定的其它信息，由于本地存储信息机制多样，例如：文件属性、注册表、数据库等，模型产生的算法标识密文和摘要可以灵活选用存储方式。

4 结束语

随着网络通讯和云计算领域的发展，各种网络终端接入，用户对信息的存储、处理和传输越来越依赖于云端，同时用户对信息的隐私等安全问题也越来越重视，包括用户身份和数据认证、完整性检测以及资源访问控制策略需求。面对海量用户和用户产生的信息，加盐信息摘要模型可以更好地抵抗有针对性的攻击行为和信息摘要碰撞，同时，使用模型进行信息摘要依赖于盐值，因而更具灵活性和保密性。

参考文献：

[1] 齐兴利. 杂凑技术和信息安全 [J]. 信息网络安全，2004(4)：57-58.

[2] Rivest R L. The MD5 message-digest algorithm [S]. RFC 1321，1992.

[3] FIPS PUB 180-1，Secure Hash Standard(SHA-1) [S]. [s. l.]：National Institute of Standards and Technology，1995.

[4] 王继敏，宋玉蓉，蒋国平. 基于消息网络的 Hash 函数构造 [J]. 计算机技术与发展，2011，21(9)：24-27.

[5] Wang X，Lai X，Feng D，et al. Cryptanalysis of the Hash Functions MD4 and RIPEMD [C]//Volume 3494 of Lecture Notes in Computer Science，Eurocrypt 2005. [s. l.]：[s. n.]，2005：1-18.

[6] Wang X Y，Yu H B. How to Break MD5 and Other Hash Functions [C]//Lecture Notes in Computer Science 3494，Eurocrypt'05. Berlin：Springer-Verlag，2005：19-35.

[7] Wang X，Yin Y L，Yu H. Finding Collisions in the Full SHA-1 [C]//Volume 3621 of Lecture Notes in Computer Science，Crypto 2005. [s. l.]：[s. n.]，2005：17-36.

[8] 乐德广，常晋义，刘详南，等. 基于 GPU 的 MD5 高速解密算法的实现 [J]. 计算机工程，2010，36(11)：154-155.

[9] 冯登国. 国内外密码学研究现状及发展趋势 [J]. 通信学报，2002，23(5)：18-26.

[10] Biham E，Dunkelman O. A Framework for Iterative Hash Functions：HAIFA [C]//NIST 2nd Hash Function Workshop. Santa Barbara：[s. n.]，2006.

[11] Aumasson J P，Meier W，Raphael C. The Hash Function Family LAKE [C]//Lecture Notes in Computer Science 5086. Berlin：Springer-Verlag，2008：36-53.

[12] 辛运帏，廖大春，卢桂章. 单向散列函数的原理、实现和在密码学中的应用 [J]. 计算机应用研究，2002(2)：25-27.

[13] 沈昌详，张焕国，冯登国，等. 信息安全综述 [J]. 中国科学(E 辑：信息科学)，2007，37(2)：129-150.

(上接第 111 页)

[4] Cowie R，Cornelius R R. Describing the emotional states that are expressed in speech [J]. Speech Communication，2003，40(1-2)：5-32.

[5] Camurri A，Poli G D，Leman M，et al. Toward communicating expressiveness and affect in multimodal interactive systems for performing arts and cultural applications [J]. IEEE Multimedia，2005，12(1)：43-53.

[6] 章国宝，宋清华，费树岷，等. 语音情感识别研究 [J]. 计算机技术与发展，2009，19(1)：92-96.

[7] 石 瑛，胡学钢，方 磊. 基于决策树的多特征语音情感识别 [J]. 计算机技术与发展，2009，19(1)：147-149.

[8] 赵 力，钱向民，邹采荣，等. 语音信号中的情感识别研究 [J]. 软件学报，2001，12(7)：1050-1055.

[9] 赵 力，钱向民，邹采荣，等. 从语音信号中提取情感特征的研究 [J]. 数据采集与处理，2000，15(1)：121-123.

[10] 王治平，赵 力，邹采荣. 基于基音参数规整及统计分布模型距离的语音情感识别 [J]. 声学学报，2006，31(1)：28-34.

[11] 赵 力，将春辉，邹采荣，等. 语音信号中的情感特征分析和识别的研究 [J]. 电子学报，2004，32(4)：606-609.

[12] 赵 力. 语音信号处理 [M]. 北京：机械工业出版社，2009：261-272.

[13] Ho S Y，Lin H，Liauh W H，et al. Orthogonal particles swarm optimization and its application to task assignment problems [J]. IEEE Transactions on Systems，Man and Cybernetics，2008，38(2)：288-298.

[14] Wei X L，Zhao Q，Zhang C J. Research on multiple index optimization method of the orthogonal test design [C]//IEEE International Conference on Computer Science and Information Technology. [s. l.]：[s. n.]，2010：224-226.

[15] 翟国富，梁慧敏，王 嗥，等. 基于正交试验设计的极化磁系统参数优化设计方法的研究 [J]. 中国电机工程学报，2003，23(10)：158-163.

[16] Liang X B. Orthogonal designs with maximal rates [J]. IEEE Transactions on Information，2003，49(10)：2468-2503.

[17] 《现代应用数学手册》编委会. 现代应用数学手册-概率论与随机过程卷 [M]. 北京：清华大学出版社，2000.

[18] 杨大利，徐明星，吴文虎. 语音识别特征参数选择方法研究 [J]. 计算机研究与发展，2003，40(7)：963-969.

[19] 陈 魁. 实验设计与分析 [M]. 北京：清华大学出版社，1996.