ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.20, No.4, April 2009, pp.902−909
doi: 10.3724/SP.J.1001.2009.03394
© by *Institute of Software, the Chinese Academy of Sciences*. All rights reserved.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

一种细粒度数据完整性检验方法*

陈 龙[1,2+], 王国胤[1,2]

[1](西南交通大学 信息科学与技术学院,四川 成都 610031)

[2](重庆邮电大学 计算机科学与技术研究所,重庆 400065)

An Integrity Check Method for Fine-Grained Data

CHEN Long[1,2+], WANG Guo-Yin[1,2]

[1](School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China)

[2](Institute of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

+ Corresponding author: E-mail: chenlong@cqupt.edu.cn

Chen L, Wang GY. An integrity check method for fine-grained data. *Journal of Software*, 2009,20(4): 902−909. http://www.jos.org.cn/1000-9825/3394.htm

Abstract: Fine-Grained integrity check for forensic data becomes an important demand of computer forensics. It will mitigate the disasterous effect on the data by some random errors or the intentional forging modification. Unfortunately, the traditional method generates a hash for every piece of small data and produces a large amount of hash data. These hash data are random data and can not be compressed in a normal way. It has a great negative impact on storing hash data and transmitting them over network. Based on the error correction coding theory, a fine-grained integrity check method, an integrity indication code, is proposed. The properties of the integrity indication code are analyzed. Combinatorial codes for one error in a group of data objects are also proposed. Hash data can be compressed hundredfold using combinatorial codes. This paper provides a fundamental support for further research on fine-grain data integrity check method and related applications.

Key words: computer forensics; hash; data integrity; error correction coding; forensic duplication

摘 要: 细粒度的数据完整性检验可以减小因偶然的错误或个别的篡改而造成的数据失效的灾难性影响,成为计算机取证的重要需求.每份数据各自生成Hash值的方法会产生大量的Hash数据.因Hash数据属于随机性数据而无法压缩,给 Hash 数据存储及网络传输带来不利影响.针对细粒度数据的完整性检验问题,提出了基于纠错编码思想的细粒度数据完整性检验方法——完整性指示码,给出了完整性指示码的若干性质.设计了指示单个错误的组合单错码,分析了该码的基本性能.结果表明,该码可以轻易地达到几百倍的压缩率.得出的结论为细粒度数据完整性检验的进一步研究及相关的应用提供了理论支持.

关键词: 计算机取证;Hash;数据完整性;纠错编码;取证复制

* Supported by the National Natural Science Foundation of China under Grant No.60573068 (国家自然科学基金); the Program for New Century Excellent Talents in University of China (新世纪优秀人才支持计划); the Natural Science Foundation of Chongqing of China under Grant Nos.CSTC2008BA2017, 2007BB2454 (重庆市自然科学基金); the Science & Technology Research Program of the Municipal Education Commission of Chongqing of China under Grant No.KJ060517 (重庆市教委科技计划项目)

Received 2008-04-03; Accepted 2008-05-19

© 中国科学院软件研究所 http://www.jos.org.cn

中图法分类号: TP311 文献标识码: A

近年来,计算机取证研究发展非常迅速,计算机取证面临的主要难题之一是海量数据处理,且目前的取证调查过程太手工化,取证时效性与取证成本都难以控制[1−3].

Hash 检验是计算机取证分析的重要手段之一.利用 MD5 和 SHA-1 等单向 Hash 函数生成的 Hash 值可以高效地检验两个数据对象(文件、数据块,不引起混淆时可简称对象)是否完全相同.单向 Hash 函数的这种能力应用于取证分析可增大自动分析的份量,如文件系统调查中的证据快速自动筛选[4,5]、海量数据相似性衡量的 Hash 包技术[6]和多分辨率相似性度量的方法[7].

在计算机取证领域,Hash 技术的一个重要应用是在取证复制的过程中计算并存储取证映像的 Hash 值,从而保证取证分析用副本的完整性,称为完整性指示.取证映像(完全复制件)的完整性不能只停留在整体是否可靠、未被修改的层面上,因为若有偶然数据变化就会影响全部数据的可用性、可信性,同时也需要从技术的角度避免以偶然错误为借口进行人为合谋篡改.所以,使用细粒度的数据完整性检验是计算机取证的必然需求,即我们需要分别判断单个文件或小数据块是否具有完整性,例如使用物理存储块大小[6].这样一来,伴随海量数据处理本身的问题,其完整性检验面临新问题——完整性检验 Hash 数据也成了大规模数据.Hash 检验数据具有随机性,无法使用数据压缩技术进行压缩,给完整性检验数据的存储和网络传输带来较大的负面影响.例如,一个 512GB 硬盘的扇区级 MD5 Hash 值将需要 16GB 的存储量,如果使用强度较高的 SHA-256,则需要 32GB.

Roussev 等人[6]在考虑衡量海量数据之间的相似性时首先意识到了 Hash 数据的大数据量问题,引入 Bloomfilter 技术将若干数据对象的 Hash 存储到一起形成一个 Hash 包——Bloomfilter.该方法的 Hash 强度大为降低,即使两个相同的 Hash 包,对应的原始数据也可能不同,不适于完整性检验.

本文借鉴纠错编码思想研究细粒度数据完整性检验新方法,实现大量 Hash 数据的压缩,保持原 Hash 检验方法的强度不变.本文首先阐述基于纠错编码理论的完整性指示及 Hash 压缩的基本思想;其次系统地阐述完整性指示编码方法,分析其主要性质;然后从基本的完整性检验及 Hash 压缩的目标出发,设计一种指示单个错误的组合指示码,组合单错指示码可以轻易地实现几百倍的压缩;最后给出结论,简要讨论未来的研究工作.

1 完整性指示与 Hash 压缩思想

1.1 纠错编码方法

信息从发送端经过通信信道到达接受端可能会出现差错,用纠错编码技术可以检测或纠正[8,9].以人们熟知的 Hamming 纠错码的[7,4,3]码为例,设 X_1,X_2,X_3,X_4 为信息元,每个信息元为一个二进制比特 0 或 1.按一致监督方程组表达监督元和信息元的监督关系为公式(1):

$$\begin{cases} X_1+X_2+X_4=X_5 \\ X_1+X_3+X_4=X_6 \\ X_2+X_3+X_4=X_7 \end{cases} \tag{1}$$

式中“+”表示模 2 相加,即异或.通信时将信息元、监督元一起发送到接收端.接收端检验时,按公式(2)计算 s_1,s_2,s_3.

$$\begin{cases} X_1'+X_2'+X_4'+X_5'=s_1 \\ X_1'+X_3'+X_4'+X_6'=s_2 \\ X_2'+X_3'+X_4'+X_7'=s_3 \end{cases} \tag{2}$$

在只出现一个比特错误的情况下,(s_1,s_2,s_3)的组合可以准确地指示出 X_1~X_7 中的任意一个错.例如,(1,1,0)表示 X_1 出错,(1,1,1)表示 X_4 出错,而(0,0,0)表示无错.由于 X_1,X_2,X_3,X_4 是二进制比特,所以特定信息位有错即可纠正.纠错编码往往使用比信息元少的监督元就可以纠正若干比特的错误,使用类似的交叉监督关系就有可能用较少的 Hash 值实现对较多的数据对象的完整性进行监督.

© 中国科学院软件研究所 http://www.jos.org.cn

1.2 基于纠错编码思想的完整性检验方法

设 X_1,X_2,X_3,X_4 表示 4 个数据对象,参照纠错编码方式,设计 Hash 检验监督关系如公式(3):

$$\begin{cases} X_1+X_2+X_4=h_1 \\ X_1+X_3+X_4=h_2 \\ X_2+X_3+X_4=h_3 \end{cases} \tag{3}$$

其中,“+”表示将数据对象连接成一个数据流,“=”表示将左端的数据流进行单向 Hash 运算,等式右端 h_1,h_2,h_3 表示 Hash 值.在需要判断数据的完整性时,按同样的方法生成测试数据的 Hash 值,判断对应的 Hash 数据是否相同(相同记为 0,不同记为 1).假设只有某一个数据对象出错(数据对象被篡改或因偶然因素发生变化等,均简称出错),则比较向量可以指示出该数据对象.例如,(1,1,0)表示 X_1 出错,(1,1,1)表示 X_4 出错.

1.3 完整性指示问题的特殊性

直接借鉴纠错编码方法有一个明显的问题.由于实际出错个数是未知的,所以(1,1,1)就有可能是 X_4 出错,也有可能是任意 2 个,乃至 3 个、4 个数据对象出错.完整性检验要求绝不能将不相同的两个数据对象误判为相同(实际相同的对象无法确认被假设为不同则可接受).如果出现错误个数无法区分的情况,只能按照都出错的情况来对待.在公式(3)的完整性检验方案里,需要少监督一个数据对象 X_4 才能达到准确指示一个错误的目标.于是可使用监督关系如下:

$$\begin{cases} X_1+X_2=h_1 \\ X_1+X_3=h_2 \\ X_2+X_3=h_3 \end{cases} \tag{4}$$

为了区别于现有的编码理论与技术,我们将基于纠错编码思想,按照一定的监督关系组合进行交叉检验,以实现完整性检验的方法称为完整性指示编码,其基本原则是不能将出错块判定为正常块.

2 完整性指示码——Hash 压缩编码

2.1 基本概念

定义 1. 令 $N=\{1,2,\ldots,n\}$为 n 个需要监督的数据对象的编号集合,$M=\{M_1,M_2,\ldots,M_m\}$为监督数据对象的 m 个 Hash 组合关系构成的集合,$M_i\subset 2^N,i=1,2,\ldots,m$.监督矩阵 A 是一个 $m\times n$ 的(0,1)矩阵,$A=[a_{ij}]_{m\times n},1\le i\le m,1\le j\le n$,其中,若 $j\in M_i$,则 $a_{ij}=1$,否则 $a_{ij}=0$.监督矩阵 A 表达了 Hash 组合与待监督对象之间的监督关系.考虑到完整性检查的意义,限定表达完整性指示关系的监督矩阵不存在相同行、相同列,也不存在全 0 行、全 0 列.

一个 6 个数据对象受 4 个 Hash 组合监督的监督矩阵实例见表 1.

Table 1 An example of check matrix

表 1 监督矩阵实例

A		n					
		1	2	3	4	5	6
m	1	1	1	0	1	0	0
	2	1	0	1	0	1	0
	3	0	1	1	0	0	1
	4	0	0	0	1	1	1

表 1 所示的监督矩阵包括了 4 选 2 的所有组合,表达了一种均匀监督关系.第 1 个 Hash 由第 1、第 2、第 4 个数据对象计算得到.

定义 2. 监督矩阵行重量:监督矩阵的第 i 行具有的 1 的个数,表达了生成第 i 个 Hash 值的数据对象个数.监督矩阵列重量:监督矩阵的第 j 列具有的 1 的个数,表达了第 j 个对象参与 Hash 计算的次数.

定义 3. 设 j_x,j_y 是监督矩阵的两列,显然,j_x,j_y 为二进制矢量.如果 j_x 等于 j_x 与 j_y 的按位或,则称 j_x 覆盖 j_y,记为 $j_x\succ j_y$.例如,$j_x=(1,0,1,1)^T,j_y=(1,0,1,0)^T$ 时有 $j_x\succ j_y$.如果有 t 列 $j_1,j_2,\ldots,j_t$,其按位或结果覆盖列 j,则称这 t 列

© 中国科学院软件研究所 http://www.jos.org.cn

$j_1,j_2,\ldots,j_t$ 共同覆盖列 j,记为 $(j_1,j_2,...,j_t)\succ j$.

定义 4. 完整性指示码.设需要检测完整性的数据对象有 n 个,若存在一种监督关系,使得生成并存储 m 个 Hash 值,在对这 n 个对象进行检测时能够准确指示任意的 t 个出错对象,而在 $n\geq t+1$ 时存在 $t+1$ 个错误的组合无法准确指示,其中单个数据对象参与 Hash 运算的次数最多为 $k(k\geq 1)$,这种监督关系称为一个完整性指示码,记为 $[n,m,t,k]$.设计这种监督关系就是设计一个编码.码的压缩率 η(也称为码率)是数据对象数 n 和使用的 Hash 值个数 m 之比:

$$\eta=\frac{n}{m} \tag{5}$$

定义 5. 利用完整性指示码 $C=[n,m,t,k]$进行完整性检验时,若实际出现的错误数 e 大于编码设计时可准确指示的错误数 t,则可能出现将正常对象判定为出错对象的情况,即指示出的出错对象大于 e,这种现象称为错误放大.由于错误对象的分布不同,实际指示错误数也可能不同.考察 e 个错误对象的所有分布,可得其平均数.实际指示错误平均数和实际出错数的比值称为错误放大率,记为 $\beta(e)$.由于码 C 能够准确指示 t 个错误,出现 $t+1$ 个错误时最能体现码的基本错误放大率特性,$e=t+1$ 时的 $\beta(e)$简记为 β,称为码 C 的错误放大率.

完整性指示码的主要性能指标有错误指示能力 t、Hash 生成计算量、压缩率以及错误放大率.

2.2 完整性指示码的性质

引理. 设 j_x,j_y 为监督矩阵中的两列且代表两个监督的对象,若 $j_x\succ j_y$,则对象 j_x 出错时无法知道对象 j_y 是否出错,只能认定对象 j_y 也已出错.

定理 1. 完整性指示码 $C=[n,m,t,k]$存在当且仅当存在 $m\times n$ 的矩阵 A.A 同时满足以下 3 个条件:

1. 矩阵 A 的列最大重量为 k;
2. 矩阵 A 的任意 t 列都不能覆盖其他的任意一列;
3. 当 $t+1<n$ 时,矩阵 A 中存在 $t+1$ 列覆盖其他的某一列.

证明:

必要性:若完整性指示码$[n,m,t,k]$存在,则由码的含义可知,码的监督矩阵 A 中列的最大重量为 k,因码 C 不能准确指示 $t+1$ 个错,所以当 $t+1<n$ 时,存在 $t+1$ 列覆盖其他的某一列.条件 1、条件 3 满足.条件 2 用反证法.假设 $j_1,j_2,\ldots,j_t,j$ 为监督矩阵中的 $t+1$ 列,分别代表 $t+1$ 个监督对象,且 $(j_1,j_2,...,j_t)\succ j$,则对象 $j_1,j_2,\ldots,j_t$ 同时出错时无法知道对象 j 是否出错,只能认定对象 j 也已出错,即出现这 t 个错时无法准确指示.这与码 C 的含义相矛盾.所以,任意 t 列都不能覆盖其他的任意一列.

充分性:已知满足 3 个条件的 $m\times n$ 矩阵 A 存在,则 A 可作为码的监督矩阵.由条件 1 可知,任意的对象参与 Hash 计算次数不会超过 k;由条件 2 可知,任选 t 列都不会覆盖其他的任意一列,则对应的 t 个对象同时出错不会影响对其他的对象的完整性判定,可准确指示 t 个错.显然,当存在 $t+1$ 列覆盖其他某列 j 时$(t+1<n)$,这 $t+1$ 个对象出错时无法知道 j 是否出错,所以码 C 不能准确指示 $t+1$ 个错.证毕. □

推论 1. 如果码的监督矩阵中存在 q 列的按位或为全 1 向量且 $q<n$,则 $t\leq q-1$.

定理 2. 完整性指示码 $C=[n,m,t,k]$的监督矩阵经任意的行交换、列交换后可构成的码 $C_1=[n,m,t,k]$,码的 η,t,k 等性能不变.

证明:显然,矩阵行交换、列交换不改变矩阵的列最大重量 k、矩阵的行、列数量 m 和 n.矩阵行交换不会改变列向量之间的覆盖关系,所以对码的错误指示能力没有影响.由定理 1 中列选择的无次序性可知,矩阵列交换对码的错误指示能力无影响,所以定理成立.证毕. □

若一个码 C_1 的监督矩阵可由另一码 C 的监督矩阵经行交换、列交换得到,此时称 C 与 C_1 等价.

定理 3. 若完整性指示码 $C=[n,m,t,k]$的压缩率 $\eta>1$,则 $k>t$.

证明:反证法,反设 $t\geq k(k\geq 1)$.为方便起见,我们称一行中位于最左的等于 1 的矩阵元素为标识元(如表 1 中的 $a_{11},a_{21},a_{32},a_{44}$),标识元刚好为 m 个.通过监督矩阵列交换使第 1 列有 k 个 1,则第 1 列有 k 个标识元,其他 $n-1$ 列都至少有一个标识元,即 $n-1\leq m-k$.

© 中国科学院软件研究所 http://www.jos.org.cn

否则,若第 j 列没有标识元,设该列重量为 k_j,$1\le k_j\le k$,该列中 k_j 个 1 共对应于 k_j 个标识元,这些标识元最多在 k_j 列上,不妨设为第 j_1 列、第 j_2 列、…、第 j_{k_j} 列,显然,这 k_j 列共同覆盖第 j 列.由定理 1 的条件 2 有 $t<k_j$,因 $k_j\le k$,所以 $t<k$,与 $t\ge k$ 矛盾.

所以,$n\le m-k+1$.此时,监督矩阵最多有 $m-k+1$ 列,即 m 行最多只能检验 $m-k+1$ 个对象($k\ge 1$),与压缩率 $\eta>1$ 相矛盾.所以 $k>t$.证毕. □

定理 4. 若两个码 $C_1=[n_1,m_1,t_1,k_1]$和 $C_2=[n_2,m_2,t_2,k_2]$的监督矩阵分别为 A_1,A_2,将 A_1,A_2 作为分块组成一个新的监督矩阵 $A=\begin{bmatrix}A_1 & 0\\ 0 & A_2\end{bmatrix}$,则矩阵 A 可构成一个新的完整性指示码 C.

$$[n,m,t,k]=[n_1+n_2,m_1+m_2,\min(t_1,t_2),\max(k_1,k_2)],$$

码 C 称为两个完整性指示码 C_1 和 C_2 的合并.

证明:矩阵行、列数量关系,最大列重量显然成立.下面来看错误指示能力 t.不妨设 $t=t_1\le t_2$,由定理 1 条件 2 可知,若 $t+1$ 列从左面 n_1 列中任选,则其中任意 t 列都不能覆盖其他的任意一列,此时可准确指示 t 个错误;若 $t+1$ 列从右面 n_2 列中任选,由 $t=t_1\le t_2$,则其中任意 t 列都不能覆盖其他的任意一列,显然也可准确指示 t 个错误;若左右两部分各选若干列,由于左、右两部分相互不会产生任何覆盖,所以仍然可以准确指示 t 个错误.由于 A_1 矩阵存在 $t+1$ 列覆盖其他某一列,所以矩阵 A 中左面 n_1 列中即存在 $t+1$ 列覆盖其他某一列,所以无法准确指示 $t+1$ 个错误.根据定理 1,新的监督矩阵 A 可构成完整性指示码 C.证毕. □

此定理表明,可以根据实际需要选取参数不完全相同的码分别监督若干对象(一般指示错误能力相同),整体上仍可看成得到统一监督.

定理 5. 若一个码 $C[n,m,t,k]$的监督矩阵 $A=\begin{bmatrix}A_1 & 0\\ 0 & A_2\end{bmatrix}$,则分别存在两个对应的码 $C_1=[n_1,m_1,t_1,k_1]$和 $C_2=[n_2,m_2,t_2,k_2]$,其监督矩阵分别为 A_1 和 A_2,称为码 C 的独立分解.其中,

$$n_1+n_2=n,m_1+m_2=m,t_1\ge t,t_2\ge t,t=\min(t_1,t_2),k_1\le k,k_2\le k,k=\max(k_1,k_2).$$

此定理描述了与定理 4 相反的过程.证明略.

定理 6. 从一个码 $C[n,m,t,k]$的监督矩阵中任意抽取 n_1 列构成矩阵 A_1,其余的列构成矩阵 A_2,则分别存在两个对应的码 $C_1=[n_1,m,t_1,k_1]$和 $C_2=[n_2,m,t_2,k_2]$,其监督矩阵分别为 A_1 和 A_2,称为码 C 的平行分解.其中,

$$n_1+n_2=n,t_1\ge t,t_2\ge t,k_1\le k,k_2\le k,k=\max(k_1,k_2).$$

证明略.定理 6 表明了由已有编码得到指示更多错误数编码的可能性. □

3 单错指示码组合设计

考虑 Hash 数据压缩需求,参考组合设计方法,设计一种针对出现单个错误情况的完整性指示码.

3.1 组合单错指示码

定理 7. m 个不同元素中任取 k 个的所有组合方案可生成 m 行 C_m^k 列的矩阵,矩阵的每列中等于 1 的元素的行号构成一种组合,该矩阵作为监督矩阵构成一个能够指示单个错误的等重量完整性指示码$[C_m^k,m,1,k]$,称为组合单错指示码(combinatorial one error integrity indication code,简称 C1eIIC).

证明:显然,矩阵有 m 行,共有 C_m^k 列,监督矩阵所有列的重量均为 k,所有行的重量均为 C_{m-1}^{k-1}.由组合关系可知,任意两种组合都不相同,即监督矩阵中任意一列都不能覆盖另一列.因任取 k 个元素的所有组合都已利用,所以两种组合至少有 $k+1$ 个不同元素,至少还有 $k-2$ 种组合的元素也在这 $k+1$ 个元素中,所以存在两列覆盖另一列.例如,$j_1=(1,2,\ldots,k)^T,j_2=(2,3,\ldots,k+1)^T,j_3=(1,3,\ldots,k,k+1)^T$ 时有 $(j_1,j_2)\succ j_3$.由定理 1,定理成立.证毕. □

因为 $m(m>1)$行的列等重关联矩阵列重量 k 有 m 种不同选法,$k=1,2,\ldots,m$,且 $C_m^k=C_m^{m-k}$,从计算代价的角度考虑,可只选取较小的 $k(k\le m/2)$.对固定的 m,k 取不同值,矩阵列数最多时可达 $C_m^{\lfloor\frac{m}{2}\rfloor}$ 列.所以,编码效率最高、压缩率

© 中国科学院软件研究所 http://www.jos.org.cn

最大的一类单错完整性指示码为

$$\left[C_m^{\left\lfloor \frac{m}{2} \right\rfloor}, m, 1, \left\lfloor \frac{m}{2} \right\rfloor\right] \tag{6}$$

定义 6. 将监督矩阵中每个监督对象 j 所对应的列矢量$(a_{1j},a_{2j},\ldots,a_{ij},\ldots,a_{mj})^T$看成是一个 m 位的二进制数,a_{1j}为最低位,a_{mj} 为最高位,则通过列交换可将监督矩阵的列从左往右按升序排列.组合单错指示码中满足此规律的监督矩阵称为规范型监督矩阵,对应的码称为规范型组合单错指示码.

3.2 Hash生成与Hash检验

计算机取证复制过程中磁盘 I/O 操作是性能的瓶颈[1].为避免重复读入数据对象或进行大量缓冲,采用规范型码的组合单错指示码$[C_m^k,m,1,k]$,其 Hash 生成采用顺序读入数据对象,渐进生成 m 个数据流并同步计算所有 m 个 Hash 值的方式.第 1 个数据对象的 Hash 编号组使用公式(7)设定,即第 1 个 Hash、第 2 个 Hash、…、第 k 个 Hash 均先计算此对象.后续其他的数据对象的 Hash 编号组依据前一编号组使用公式(8)迭代计算得到.

$$(r_1^{(1)},r_2^{(1)},\ldots,r_k^{(1)})=(1,2,\ldots,k) \tag{7}$$

$$(r_1^{(j+1)},r_2^{(j+1)},\ldots,r_k^{(j+1)})=\begin{cases}(r_1^{(j)}+1,r_2^{(j)},\ldots,r_k^{(j)}), & r_1^{(j)}+1<r_2^{(j)}\\ (1,r_2^{(j)}+1,r_3^{(j)},\ldots,r_k^{(j)}), & \begin{cases}r_1^{(j)}+1=r_2^{(j)}\\ r_2^{(j)}+1<r_3^{(j)}\end{cases}\\ \ldots, & \ldots\\ (1,2,\ldots,k-1,r_k^{(j)}+1), & \begin{cases}r_{l-1}^{(j)}+1=r_l^{(j)},l=2,3,\ldots,k\\ r_k^{(j)}<m\end{cases}\end{cases} \tag{8}$$

若完整性检验时 m 个 Hash 值中出现 $r(r<k)$个无法匹配,表明只是存储的 Hash 值出现错误,所有数据对象的完整性均有保证.

若 m 个 Hash 值中刚好有 k 个无法匹配,其编号从小到大依次为 $r_1,r_2,\ldots,r_k$.,则刚好有一个数据对象的完整性无法保证,判断为出错.依据规范型监督矩阵的规律,考虑相对于第 1 个数据对象的偏移,第 i 个编号 r_i 造成的数据对象偏移量为 $C_{r_i-1}^i$ (规定 $C_r^k=0,k>r$).所以出错数据对象的编号为

$$j=1+\sum_{i=1}^{k}C_{r_i-1}^i \tag{9}$$

若 m 个 Hash 值中有 $r(r>k)$个无法匹配,则有 C_r^k 个对象出错,可由 r 个不同编号中选 k 个编号按公式(9)计算出所有出错对象编号.

3.3 组合单错指示码性能分析

组合单错指示码$[C_m^k,m,1,k]$的压缩率 $\eta=\frac{1}{k}C_{m-1}^{k-1}$,对于较小的 m 和 k,各种不同组合单错码的压缩率如图 1 所示.

由图 1 可见,随着 m 和 k 的增长,码的压缩率增长得很快.整体而言,C1eIIC 码的压缩率很高,如[495,12,1,4]的压缩率为 41.25,而[4845,20,1,4]的压缩率达到 242.25.

考察出错放大率.C1eIIC 码的一个数据对象 b 出错会导致 k 个 Hash 值不能匹配,根据所有其他数据对象与 b 共享 Hash 值的个数 $i(0\le i<k)$,可将其他数据对象分为 i 类,第 i 类的数据对象个数为 $C_k^i\cdot C_{m-k}^{k-i}$.相对于出错对象 b,如果另一个出错数据对象在第 i 类,则共有 $2\times k-i$ 个 Hash 值不能匹配,此时,实际判断为出错的数据对象有 C_{2k-i}^k 个.所以组合单错指示码$[C_m^k,m,1,k]$的错误放大率为

$$\beta=\frac{1}{2\cdot(C_m^k-1)}\sum_{i=0}^{k-1}(C_{2k-i}^k\cdot C_k^i\cdot C_{m-k}^{k-i}) \tag{10}$$

对于较小的 m 和 k,C1eIIC 码的错误放大率如图 2 所示.

© 中国科学院软件研究所 http://www.jos.org.cn

由图 2 可见,组合单错指示码 k=2,3 时错误放大率较小,而 k=4,5 时错误放大率较大,增长也较快.例如,码[495,12,1,4],[792,12,1,5]的错误放大率分别为 15.61 和 32.59.

从以上分析结果可以看出,组合单错指示码的重要特性是压缩率很高,但指示错误数较少,错误放大率较高.

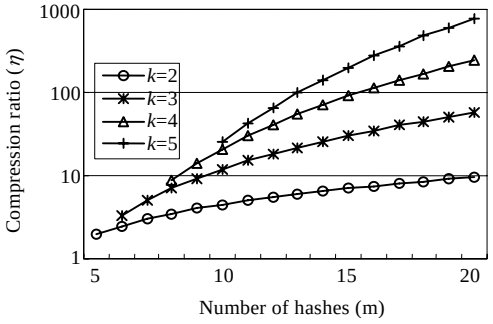

Fig.1 Compression ratio of various C1eIIC codes

图 1 组合单错指示码的压缩率

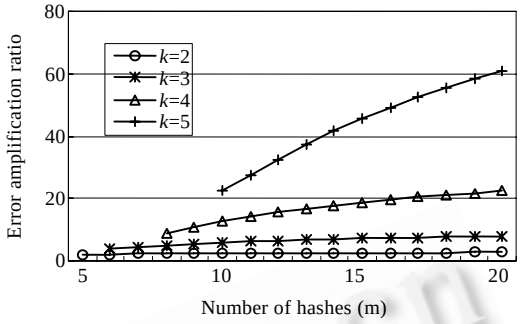

Fig.2 Error amplification ratio of various C1eIIC codes

图 2 组合单错指示码的错误放大率

3.4 完整性指示码的应用分析

数据完整性的保证是通过预先生成并存储完整性检验数据,在进行检验时按相同的方法重新生成测试数据的完整性检验数据,把两份完整性检验数据进行比较来判断数据是否具有完整性.当组合单错指示码生成 m 个 Hash 值时,每个数据对象需要参与 Hash 计算 k 次.即 Hash 计算量是原来的 k 倍.由于计算机取证复制过程中磁盘 I/O 操作占据主要的时间,是影响性能的瓶颈,所以数据对象参与多次 Hash 计算对性能不产生实质性影响.尽管如此,仍建议在实际应用中只选用较小的 k 值,如 2,3,4.同时,因为采用了同步计算所有 Hash 的方式,在 Hash 生成环节只读入数据 1 次,所以组合单错指示码不增加读入数据的额外开销.

4 结论和未来的工作

本文提出了细粒度取证对象完整性检验问题和基于纠错编码思想的完整性检验方法——完整性指示编码,分析了码的主要性质.给出了检测一组取证对象中出现一个错误时的组合完整性指示码,分析了其基本性能,提供了在低出错率的条件下实现 Hash 数据大幅度压缩的方法.本文的结论为完整性检验的进一步研究以及 Hash 数据压缩、分组交叉完整性检验相关的应用(如证据、多证据完整性指示存储分离)提供了理论支持.

计算机取证在实践中还有指示多个错误等其他一些实际需求,在性能方面往往需要考虑特定场合并加以折衷,这些问题需要在进一步的研究中逐步加以解决.下一步的研究包括用现有基本理论分析已有的各种纠错编码在细粒度完整性指示方面的可用性,进一步完善细粒度完整性指示编码理论,设计实用的可指示多个错误的完整性指示码以及开发取证工具.

References:

[1] Golden G. Richard III, Roussev V. Next-Generation digital forensics. Communications of the ACM, 2006,49(2):76−80.

[2] Wang L, Qian HL. Computer forensics and its future trend. Journal of Software, 2003,14(9):1635−1644 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/14/1635.htm

[3] Ding LP, Wang YJ. Study on relevant law and technology issues about computer forensics. Journal of Software, 2005,16(2): 260−275 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/16/260.htm

[4] Kornblum J. Identifying almost identical files using context triggered piecewise hashing. Digital Investigation, 2006,3(s1):91−97.

[5] Chen L, Wang GY. An efficient piecewise hashing method for computer forensics. In: Luo Q, Gong MM, Xiong F, Yu F, eds. Proc. of the Int'l Workshop on Knowledge Discovery and Data Mining. Adelaide: IEEE Comupter Society, 2008. 635−638.

© 中国科学院软件研究所 http://www.jos.org.cn

[6] Roussev V, Chen YX, Bourg T, Richard III GG. Md5bloom: Forensic filesystem hashing revisited. Digital Investigation, 2006, 3(s1):82−90.

[7] Roussev V, Richard III GG, Marziale L. Multi-Resolution similarity hashing. Digital Investigation, 2007,4(s1):105−113.

[8] Jin F, Chen Z. Combinatorial Coding Theory and its Applications. Shanghai: Shanghai Scientific & Technical Publishers, 1995 (in Chinese).

[9] Bose R. Information Theory, Coding and Cryptography. Beijing: McGraw-Hill Education Co., China Machine Press, 2003.

附中文参考文献:

[2] 王玲,钱华林.计算机取证技术及其发展趋势.软件学报,2003,14(9):1635−1644. http://www.jos.org.cn/1000-9825/14/1635.htm

[3] 丁丽萍,王永吉.计算机取证的相关法律技术问题研究.软件学报,2005,16(2):260−275. http://www.jos.org.cn/1000-9825/16/260.htm

[8] 靳番,陈志,编著.组合编码原理及应用.上海:上海科学技术出版社,1995.

陈龙(1970－),男,重庆人,博士生,副教授,CCF 高级会员,主要研究领域为计算机取证,网络安全,智能信息处理.

王国胤(1970－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为知识获取,粗糙集,粒计算,知识技术.

© 中国科学院软件研究所 http://www.jos.org.cn