(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 115051789 A
(43)申请公布日 2022.09.13

(21)申请号 202210670443.6

(22)申请日 2022.06.15

(71)申请人 道和邦(广州)电子信息科技有限公司

地址 510440 广东省广州市白云区嘉禾街鹤龙二路96号粤旺大厦C栋305室

(72)发明人 陈书增

(51)Int.Cl.

H04L 9/06(2006.01)

H04L 9/40(2022.01)

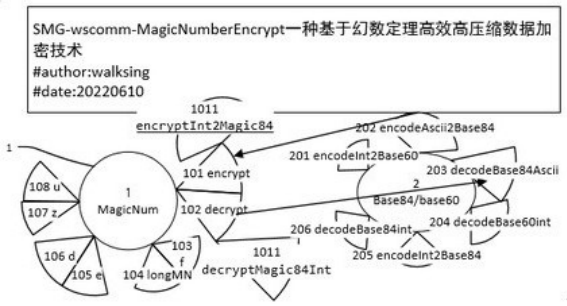
权利要求书5页 说明书9页 附图2页

(54)发明名称

SMG-wscomm-MagicNumberEncrypt一种基于幻数定理高效高压缩数据加密技术

(57)摘要

SMG-wscomm-MagicNumberEncrypt一种基于幻数定理高效高压缩数据加密技术:本发明为基于幻数定理推导出一系列公式,可实现对任意自然数通过与基幻数发生作用产生新的幻数,通过幻数运算将自然数乘法改变为幻数以10为底进行高低位移位加法运算,直接提高运算效率几个数量级,本发明为颠覆性技术超越传统计算理论,甚至改变数学乘法基础;本发明主要用于动态令牌加密编码传输;以及航天航空领域的大量观测大数进行压缩,结合Base84算法(参同期专利)可实现高安全性,高可靠性信息保密压缩传输;高倍空间压缩编码,高效率加密解密技术为超算力输出,超分布式存储输出必放一异彩。



CN 115051789 A

1.一种基于幻数:142857推理的定理,其特征如下:

定义:幻数,任何自然数与142857相乘积得到的自然数即为幻数;

幻数依据下列表格进行排列:

```
final static long[] MN={
        142857, 285714, 428571, 571428, 714285, 857142,999999,999999,//7 7x1
        1142856,1285713,1428570,1571427,1714284,1857141,1999998,//14 7x2
        2142855,2285712,2428569,2571426,2714283,2857140,2999997,//21 7x3
        3142854,3285711,3428568,3571425,3714282,3857139,3999996,//28 7x4
        4142853,4285710,4428567,4571424,4714281,4857138,4999995,//35 7x5
        5142852,5285709,5428566,5571423,5714280,5857137,5999994,//42 7x6
        6142851,6285708,6428565,6571422,6714279,6857136,6999993,//49 7x7
        7142850,7285707,7428564,7571421,7714278,7857135,7999992,//56 7x8
        8142849,8285706,8428563,8571420,8714277,8857134,8999991,//63 7x9
        9142848,9285705,9428562,9571419,9714276,9857133,9999990,//70 7x10
         10142847,10285704,10428561,10571418,10714275,10857132,10999989,//77 7x11
        11142846,11285703,11428560,11571417,11714274,11857131,11999988 //84 7x12 MAP base84
       ...
}
```

第1行0=142857,1=142857*1,..6=142857*6,7=142857*7 第2行8=142857*8,..

第n位=142857*n;

经过周密计算经过如上表格组合排列可推导如下:

1.3 幻数定理:

定义:幻数,即以142857为倍数或除数的自然数.

令:f0=142857为基幻数,n为>=0自然数,f(n)即n的幻数.

1.3.1 定理1: 任何一个自然数可通过与基幻数相乘得到另外一个幻数即:

n*f0=f(n),并由此易证f(m+n)=f(m)+f(n),m,n为自然数>7.

1.3.2 定理2: 对于幻数当n>=0,n<7 ,满足于f(n)=(n+1)*f0;且f(7)=f(6)=999999.

1.3.3 定理3: 俩个大的自然数乘法可转为俩个较大的幻数加法运算,因计算机加法效率比乘法效率高,故任意俩个大的自然数相乘均可通过幻数运算转为加法运算使得计算效率成倍提高.

即:f(n)=((long)((n-1)/7))*1000000+f((n-1)%7)-(long)((n-1)/7),(n>7,n→∞);

或:f(n)=((long)((n-1)/7))*(7*f0)+f((n-1)%7),(n>7,n→∞)。

2.一种基于幻数的加密解密算法,其特征如下:

2.1 基于 1.3.1,1.3.2,1.3.3, 利用幻数定理可将任意俩个自然数转为幻数进行存储并加密解密,实现将传统乘法转为幻数加法,以实现计算机的快算、超算功能,其算力可类优于传统珠算;

2.2 基于2.1可实现信息安全加密存储与信息解密,以实现通信信息安全基础服务,其特征如下:

2.2.1根据2.1推导加密、解密算法公式,特征如下:

MagicNumberEncrypt=

令 f(0)=f0=142857~=1000000*(1/7)=142857.142857142857...;

* f02=f0*f0=20408122449L;

* f(0)=142857

* f(1)=f(0)*2=285714

* f(2)=f(0)*3=428571

* f(3)=f(0)*4=571428

* f(4)=f(0)*5=714285

* f(5)=f(0)*6=857142

* f(6)=f(0)*7=999999

* f(7)=f(0)*7=999999

*

* 令 k∈{0,999999} 作为加密密钥的长整数

* n>0自然数

* 当n=0,令f(0)=f0=142857;当n=7令f(7)=f(6)=f(0)*7=999999;

* (基于1.3.1)依据定理1:f(n)=n*f0,n>0 自然数;

* (基于1.3.1)依据定理1:f(n)的逆函数n=longMN(x)=x/f0,n>7,0>=n<7, return n-1;

*

* R:加密函数e(n)

* 令e(n)=7*f(n)+f(k/7)+f0*f(k%7);

* =>7*n*f0+(k/7)*f0+f0*(k%7)*f0;

* =>(7n+k/7+f0*(k%7))*f0;

* =>7n*f0+k*f0/7+(k%7)*f0²;

=>(k%7)f0²+(7n+k/7)*f0

=>ay²+by;

(a=k%7,b=7n+k/7,y=f0);1)

* =>x=(k%7)f0²+(7n+k/7)*f0;2)

*

* R:解密函数n=d(x), x=e(n)

* 令d(x)=(x-f(k/7)-f0*f(k%7))/7/f0

=>(x-f(k/7)-f0*f(k%7))/f(7);

* =>(x-(k/7)*f0-f0²*(k%7))/(7f0);

=>-(k%7/f(7))*f0²-(k/(7²*f0))*f0+x/f(7); =>-cy²-dy+x/f(7) (c=k%7/f(7),d=k/f(49));1)

* n=>x/(7f0)-(k/7²)-f0*(k%7)/7;2)

{证明法1)::if e(n)=x; then d(x)=n,假定正确,证明如下:

由R:e(n)得,

$x=e(n)=ay^2+by$; (a=k%7,b=7n+k/7,y=f0);

由R:d(x)得

$n=d(x)=-cy^2-dy+x/f(7)$; (c=k%7/f(7),d=k/f(49));

代入x,a,b

$=>n=d(x)=>-cy^2-dy+((k\%7)f0^2+(7n+k/7)*f0)/f(7)$;

代入c,d

$=>n=-(k\%7/f(7))y^2-(k/f(49))y+((k\%7)y^2+(7n+k/7)*y)/f(7)$;

$=>n=-(k\%7/f(7))y^2-(k/f(49))y+((k\%7)/f(7))y^2+((7n+k/7)/f(7))*y$;

合并y^2,y

$=>n=(-(k\%7/f(7))+((k\%7)/f(7)))y^2+(-(k/f(49))+(7n+k/7)/f(7))y$;

$=>n=(0)y2+(-(k/f(49))+(7n+k/7)/f(7))y$;

$=>n=(-k/f(49)+7n/f(7)+k/f(49))y$;

$=>n=(7n/f(7))y$;

$=>n=(7n/7*f0)y$;

$=>n=(n/f0)y$;

as y=f(0)=f0;

$=>n=(n/f0)*f0$;

=>n=n 成立;

证毕.

}

R1:x*f(n)=n*f(x);

R2:d(x)=n,e(n)=x;

*

*{证明法2: if e(n)=x; then d(x)=n; 假定正确,证明如下:

*令

* R1) $x=e(n)=(k\%7)f0^2+(7n+k/7)*f0$;

* R2) $n=d(x)=x/(7f0)-(k/7^2)-f0*(k\%7)/7$;

*

* 由R2:d(x)

*$=>n=d(x)=x/(7f0)-(k/7^2)-f0*(k\%7)/7$;3)

$=>x=(n+k/7^2+(k\%7)*f0/7)*(7f0)$;

*

* $=>x=7n*f0+k*f0/7+(k\%7)*f0^2$;

由R1:$x=e(n)=(k\%7)f0^2+(7n+k/7)*f0$;

* $7n*f0+k*f0/7+(k\%7)*f0^2=(k\%7)f0^2+(7n+k/7)*f0$;

$=>7n*f0+k*f0/7=7n*f0+f0*k/7$;

$=>k*f0/7=f0*k/7$;

=>k*f0=f0*k 乘法交换律成立;

证毕.

* }

2.2.2 基于2.2.1

通过R2:3)

知:n=d(x)=x/(7f0)-(k/7²)-f0*(k%7)/7;x是密文,k是密钥

k,n 均不为0的情况下,每个二元一次方程都有无数对方程的解,导致求解困难,反证加密算法安全性.

2.2.3 基于2.2.1

通过R1:1)

知:x=ay²+by=>ay²+by-x=0如果x,y互换=>y=ax²+bx; (a=k%7,b=7n+k/7,x=f0,a!=0);

由y=ax²+bx ,a=k%7,b=7n+k/7,x=f0,由公式知,当k值变化一点,y值变化很大;

这样很小的k的变化会引起很大的加密数据的改变,y相对于n值变化就越大,信息失真度就越明细,原始信息保密性就越强大。

3.基于2信息加密、解密算法结合Base84/60编码(查前专利)可实现网站令牌的高倍高效压缩加密技术方案,其特征如下:

3.1 用户数据令s=user.p.toStr(),s为以&…= 为格式的用户基础数据,其来源于上次登录成功后数据调用之AFS(可迭代分布式存储引擎),明文数据长=1613字节;基于本发明3编码Base84编码再进行加密,令e=MagicNumberEncrypt.encrypt(k,s);得到密文e;e密文长2361字节,耗时:10812751ns,解密:9239864ns;也即加密压缩比146%,远小于传统的200%压缩比;AES:压缩后密文长度:2200字节,压缩比:136%,AES:耗时:714830523ns,AES解密:2755762ns,压缩比对比:146:136,加密速度:72.3:1,解密速度:0.27:1,安全性:前者比后者更强;

3.2 基于2对信息加密过程,需要进行幻数计算,因幻数存储空间较大,利用密文被幻数整除原理需要对密文进行压缩处理,z函数由此诞生,产生压缩码对压缩码进行base84/60编码,再合并发送;

3.3 基于2,3.2对信息解密过程MagicNumberEncrypt.decrypt(k,s),先对信息进行Base84/60解码,再合并,再进行逆还原通过u函数对压缩码进行解码还原为大幻数,再对幻数进行d(k,s)解密。

4.基于3可实现网站令牌的无登录认证技术:用户基本信息被加密,存储在cookie,user键,总长约2.3kb,不超过4K,用户端无密钥,且密钥不传输,仅在服务端进行解密令牌,通过OTP_SN口令可实现将流密钥动态刷写于令牌,通过浏览器及时交互往返传输于用户端与服务器端,OTP_SN1分钟自动过期,其密钥可通过IPSC/IP技术(查前专利)通过js无痕代码经过计算求得OTP_SN,js端拿到OTP_SN可实现对url参数进行客户端sign签名提交数据,以保护数据安全,其表现为当登录端,输入账户,密码,鼠标离开密码输入框失去焦点时,密码,自动调整为较初始值更长的密文;

OTP_SN时效过期特征:表现在当在提交订单支付前,服务端提示:“签名无效,返回刷新重新提交”类提示,此时因为web端OTP_SN已经过期,返回前一步重新刷新再提交即可通过,因为服务端Msession.genOTP_TLSP算法会自动产生基于时间不可预测LSP公平算法(查前

专利)密钥发生器,而系统在解密token令牌前会自动读取最新的密钥SN并加密推入令牌并2次加密。

5.基于4可实现在https/http协议非安全协议实现轻量级较为可靠的安全保密通信,基于4可实现签名防篡改技术。

6.基于1,2幻数定理,可广泛应用于军用,民用,航空,航天大数超算,以及信息安全领域,其表现为较快的运算速度,较快的信息加密速度,以及较高的信息安全破解难度。

7.基于4,5可广泛应用于离线超算与离线存储,离线交易结算,离线钱包功能,具体表现所为离线是指常时间的不操作AFS存储器,所有的计算,与交易结算可缓存在cookie端加密存储,可在多个分布微交易后一次性push到AFS存储器进行数据同步更新操作,这样可减少数据存储的频繁交互,减轻服务端压力。

8.基于1幻数,其思路类比珠算,可改变传统10进制乘法,通过幻数的高低位移位运算可代替传统的10进制乘法运算,使得运算效率提示60倍。

SMG-wscomm-MagicNumberEncrypt一种基于幻数定理高效高压缩数据加密技术

技术领域

[0001] 本发明为一种新型颠覆性超算技术,类似珠算代替乘法计算,通过研究幻数142857,据传最早被科学家发行于古埃及金字塔,幻数又名金字塔数,跑马灯数在1-6间自然数与其相乘所得数始终围绕在142857轮转,幻数本质是1/7的小数部分无限循环的循环节,本发明者经过反复推导计算发现幻数定理,并加以推衍应用于计算机领域信息安全领域高效率高安全性的信息加密、解密公式,通过数学论证其可靠性合理性。

[0002] 本项发明,为SMG-VME可迭代分布式操作系统,SMG-VME-AFS可迭代分布式存储系统延伸的价值,iSBS/mbs可迭代超级商城动态令牌技术的应用基础,主要应用于信息的高倍编码,高效率的加密应用。

[0003] 本项发明,为SMG-VME系列下软件工程实现的社会使命,其目标为解决超算,超存储输出,以及信息的安全传输,安全获取。

[0004] 本项发明

专利审查:

截至2022-06-01网络搜索暂无同案例

背景技术

[0005] 本发明主要应用在工业互联网,万物智能互联,分布式计算,万物互联数据信息的加密交换,信息的高速运算。本发明最初在设计一种可靠的加密算法,高压缩高安全性信息加密/解密技术,传统Base64位编码不能有效的压缩数字,最初在设计iSBS/mbs可迭代超级商城的订单号问题,订单号长度通常设定15位数字比较合理15位数字刚巧是时间精巧到秒的纯数字,如果订单号太短,则在大用户并发下会引发抢单冲突,即订单号在相同的时间单位秒下生成订单号流水相同,引发不可预料的错误,而一个购物车下会挂几个商家的产品,在用户一次结算购物车事务会引发多个商家的订单支付问题,这涉及到批量订单消费问题,解决思路有俩个 a)通过建立购物订单支付与商家订单建立订单映射表,可实现1:m,m不限量,但在查询支付结算过程势必要进行与订单映射表关联,多一次数据库循环查询,增加数据库负担.

b)将购物车多个商家的订单号通过字符串逗号分隔多个商家放在支付订单表后,但通常总厂有限制长度不会超过250,此方案优点是避免关联查询,查询速度快,对数据库减轻负担,缺点:支持的订单号数量有限制,该方案能存储最多250/(15+1)=15个商家订单号. 总结:a),b) 单个订单号信息不能短,可容纳订单数量尽可能多,查询效率尽可能快.在此背景下,要求设计一种算法能将15位数字压缩到8位,且信息不丢失;故研发了Base84/60进制编码方案,该方案经过实践测试可压缩到8位,基于b)方案单次购物车下单可容纳不同商家数量为250/(8+1)=27个扩容接近1倍,一次购物27个不同商品基本可满足用户需求.

在实现了Base84/60 Base60主要用来对数字领域的压缩且表示字符敏感领域,如数据库经常存储,其标识符:A-Y,a-y,0-9 组合60字符,进位用z表示.对非数据库场景用

Base84压缩更省空间,且后者用来对ascii码中文混合编码进行重编码,效率比base64要高,具体(查专利Base84/60).有了Base84/60高效压缩编码,首先解决了购物订单压缩问题,后又面临新的技术问题,本发明最初考虑用于iSBS/mbs可迭代超级商城的无登录动态令牌加密系统,因为该系统是无登录技术,要求第一次登录把用户基本信息加密压缩存储于cookie,后续每次请求带令牌,然后系统解密,因为是用户频繁操作,要求加密/解密速度快,响应时间短.

无登录动态令牌,所谓动态是因含有OTP_TLSP算法的密钥发生器1分钟过期,重新生成密钥写入cookie,用户基本信息user.p.toStr()以&..=为组合大约有1600字节长度;经过本算法加密并结合Base84/60压缩编码产生密文约2300字节,较AES加密并Base64编码,密文长约10%,但加密效率速度较后者提升至72倍,经过推算其安全性亦大大提高,高安全的原因是经过推算解密n的公式为二元一次方程,对任意2元一次方程ax+by+c=0(a、b≠0)有无数解,基于此原理导致破解困难,本发明支持各种数据组合加密,即密文之中含密文;比如实际应用中将user属性数据加密存储在cookie,属性内有esn即把OTP_SN以AES加密存储放入user.外层加密统一用本发明MagicNumberEncrypt.encrypt加密;外层好比防火墙.里边又像一个小世界可用自己喜欢的算法加密,经过实践已经成功并实施于iSBS/mbs动态令牌无登录认证技术.

附图说明

[0006] *图1是MagicNumberEncrypt 幻数加密类的逻辑函数图;*

1MagicNumberEncrypt幻数加密

101 encrypt 加密

102 decrypt 解密

1011encryptInt2Magic84

1012 decryptMagic84Int

103 f 长整数转幻数函数

104 longMN 幻数转长整数 103逆运算

105 e 单元幻数加密

106 d 单元幻数解密105逆运算

107 z 经过105 e运算后压缩码

108 u 经过106 d后解压缩码,107的逆运算

图2是MN是基幻数与自然数的乘积表按7进制前84为排列组合;

图3幻数加密过程调用逻辑图;

图4是幻数解密过程调用逻辑图.

加密调用逻辑:参图1,3

图3,101 encrypt->输入k,s ,k为密钥,s为任意字符串(可包含中文)->图3,10101 s=Base84.encodeAscii2Base84(s)进行base84 encode编码->遍历s

调图3,101020 MagicNumberEncrypt.encryptInt2Magic84(k,s)->调用图3,10102 切割分组转为长整数再进行x=e(k,1)加密逻辑运算,e调用图3,101021 f(x)转为幻数,生成幻数,由图3,10103 z函数进行压缩,并封装,完成后调图3,10104

Base84.encryptInt2Base84 返回加密的Base84 ascii码 加密完成.

解密调用逻辑:参图1,4

图4,102 decrypt ->输入k,s=密文,k为密钥,s为基于Base84 ascii编码的密文;->调图4,10201 MagicNumberEncrypt.decryptMagic84Int ->调图4,102011 S=Base84.decodeBase84Int 将base84 转为Int串->然后遍历

->调图4,102012 t=u(t) t分组密文解压->图4,102013 x=d(k,l) 单元幻数解密->图4,1020131 longMN(x) =f(n)的逆函数 n=longMN(f(n)) 解出n并封装输出->图4,10204 Base84.decodeBase84Ascii 解密明文,解密结束.

发明要义:

1 一种基于幻数:142857推理的定理,其特征如下:

1.1 定义:幻数,任何自然数与142857相乘积得到的自然数即为幻数;

1.2 幻数依据下列表格定义进行排列:

```
finalstaticlong[] MN={
        142857, 285714, 428571, 571428, 714285, 857142,999999,
999999,//7 7x1
        1142856,1285713,1428570,1571427,1714284,1857141,1999998,//14
7x2
        2142855,2285712,2428569,2571426,2714283,2857140,2999997,//21
7x3
        3142854,3285711,3428568,3571425,3714282,3857139,3999996,//28
7x4
        4142853,4285710,4428567,4571424,4714281,4857138,4999995,//35
7x5
        5142852,5285709,5428566,5571423,5714280,5857137,5999994,//42
7x6
        6142851,6285708,6428565,6571422,6714279,6857136,6999993,//49
7x7
        7142850,7285707,7428564,7571421,7714278,7857135,7999992,//56
7x8
        8142849,8285706,8428563,8571420,8714277,8857134,8999991,//63
7x9
        9142848,9285705,9428562,9571419,9714276,9857133,9999990,//70
7x10
         10142847,10285704,10428561,10571418,10714275,10857132,
10999989, //77 7x11
         11142846,11285703,11428560,11571417,11714274,11857131,
11999988 //84 7x12 MAP base84
      //…
    }
```

第1行0=142857,1=142857*1,..6=142857*6,7=142857*7 第2行8=142857*8,..

第n位=142857*n;

经过周密计算经过如上表格组合排列可推导如下:定理

1.3 幻数定理:

定义:幻数,即以142857为倍数或除数的自然数.

令:f0=142857,n为>=0自然数,f(n)即n的幻数.

定理1: 任何一个自然数可通过与幻数相乘得到另外一个幻数即:

n*f0=f(n),并由此易证f(m+n)=f(m)+f(n).

定理2: 对于幻数当n>=0,n<7 ,满足于f(n)=(n+1)*f0;且f(7)=f(6)=999999.

定理3: 俩个大的自然数乘法可转为俩个较大的幻数加法运算,因计算机加法效率比乘法效率高,故任意俩个大的自然数相乘均可通过幻数运算转为加法运算使得计算效率成倍提高.

即:f(n)=((long)((n-1)/7))*1000000+f((n-1)%7)-(long)((n-1)/7),(n>7,n→∞);

或:f(n)=((long)((n-1)/7))*(7*f0)+f((n-1)%7),(n>7,n→∞).

```
/**推理:
 * if{
 * f0=f(0)=142857≈1000000*(1/7)=142857.142857142857...;
 * by 定理2:f(n)=f(0)*(n+1)  0<=n<7;
 *
 * f(1)=f(0)*2=285714
 * f(2)=f(0)*3=428571
 * f(3)=f(0)*4=571428
 * f(4)=f(0)*5=714285
 * f(5)=f(0)*6=857142
 * f(6)=f(0)*7=999999
 * f(7)=f(0)*7=999999
 * }
 * 参考 MN表
 * f(8)=(1)*1000000 + f(0)-1;=>f(8)=1*1000000+f(m)-1;
 * f(15)=(2)*1000000 + f(0)-2;
 *
 * f(22)=(3)*1000000 + f(0)-3;=>f(22)=3*1000000+f(m)-3;
 * ...
 * 猜想:
 *
 * f(n)=((long)((n-1)/7))*1000000+f((n-1)%7)-(long)((n-1)/7);
 * =>((long)((n-1)/7))*999999+f((n-1)%7);
 *
```

* {证明:证明定理1=定理3即

n*f0=f(n)=((long)((n-1)/7))*1000000+f((n-1)%7)-(long)((n-1)/7)=n*f0,(n>7,n→∞).

* 由定理3

假设正确则有

* f(n) =>((long)((n-1)/7))*1000000+f((n-1)%7)-(long)((n-1)/7);(n>7,n→∞)

* =>((long)((n-1)/7))*(1000000-1)+f((n-1)%7);

* =>((long)((n-1)/7))*(999999)+f((n-1)%7);

* =>((long)((n-1)/7))*(7*f0)+f((n-1)%7);

*注:此项7与分母7暂时不能约去因为(long)((n-1)/7) 在java语言表示取整之意,抛弃小数部分,必须分优先级计算.

…

n in [7-97] 均计算正确此忽略

* if n=98

=>f(98)=(13)*7*f(0)+f(97%7)=91*f0+f(6)=91*f0+7f0=98f0;

* if n=99

=>f(99)=(98/7)*7*f(0)+f(98%7)=99*f0;

* if n=100

* f(100)=(99/7)*7*f(0)+f(99%7)=>14*7f0 +f(1)=>98f0+2f0=100*f0;

* f(101)=(100/7)*7*f(0)+f(100%7)=14*7f0+f(2)=98f0+3f0=101*f0;

* ...

*

* f(n)=((long)(n-1)/7)*7f(0)+f((n-1)%7);(n>7,n→∞)

* 设当n-1->x,且满足x被7整除,令x=7m

代入公式则有

f(x+1)=((long)(n-1)/7)*7f(0)+f((n-1)%7);(n>7,n→7m+1)

=>f(x+1)=((long)(7m+1-1)/7)*7f(0)+f((7m+1-1)%7);(n>7,n→7m+1)

=>f(x+1)=((long)(7m)/7)*7f(0)+f((7m+1-1)%7);(n=7m+1)

=>f(x+1)=7m*f0+f(0);

=>f(x+1)=7m*f0+f0;

as x=7m

=>f(7m+1)=7m*f0+f0;

由定理1 f(n)=n*f0=>f(n)=f(n-1)+f0=>f(m+n)=f(m)+f(n),代入上式:

=>f(7m+1)=f(7m)+f0;

=>f(7m)+f0=f(7m)+f0;

=>f(7m)=f(7m);

7m->换成n

=>f(n)=f(n) 亦成立

* f(n)=((long)(n-1)/7)*7f(0)+f((n-1)%7);(n>7,n→∞)

=>当n→∞时,n-1 ->9999…, f(n)≈(n-1)*f(0)+f(0);

=>f(n)=n*f(0)=n*f0;

* 证毕.

*

* }

*

2一种基于幻数的加密解密算法,其特征如下:

2.1 基于 1.3.1,1.3.2,1.3.3, 利用幻数定理可将任意俩个自然数转为幻数进行存储并加密解密,实现将传统乘法转为幻数加法,以实现计算机的快算、超算功能,其算力可类优于传统珠算;

2.2 基于2.1可实现信息安全加密存储与信息解密,以实现通信信息安全基础服务,其特征如下:

2.2.1根据2.1推导加密、解密算法公式,特征如下:

MagicNumberEncrypt=

令 f(0)=f0=142857~=1000000*(1/7)=142857.142857142857...;

* f02=f0*f0=20408122449L;

* f(0)=142857

* f(1)=f(0)*2=285714

* f(2)=f(0)*3=428571

* f(3)=f(0)*4=571428

* f(4)=f(0)*5=714285

* f(5)=f(0)*6=857142

* f(6)=f(0)*7=999999

* f(7)=f(0)*7=999999

*

* 令 k∈{0,999999} 作为加密密钥的长整数

* n>0自然数

* 当n=0,令f(0)=f0=142857;当n=7令f(7)=f(6)=f(0)*7=999999;

* (基于1.3.1)依据定理1:f(n)=n*f0,n>0 自然数;

* (基于1.3.1)依据定理1:f(n)的逆函数n=longMN(x)=x/f0,n>7,0>=n<7, return n-1;

*

* R:加密函数e(n)

* 令e(n)=7*f(n)+f(k/7)+f0*f(k%7);

* =>7*n*f0+(k/7)*f0+f0*(k%7)*f0;

* =>(7n+k/7+f0*(k%7))*f0;

* =>7n*f0+k*f0/7+(k%7)*f0²;

=>(k%7)f0²+(7n+k/7)*f0

=>ay^2+by;

(a=k%7,b=7n+k/7,y=f0);1)

* =>x=(k%7)f0²+(7n+k/7)*f0;2)

*

* R:解密函数n=d(x), x=e(n)

* 令d(x)=(x-f(k/7)-f0*f(k%7))/7/f0

=>(x-f(k/7)-f0*f(k%7))/f(7);

* =>(x-(k/7)*f0-f0²*(k%7))/(7f0);

=>-(k%7/f(7))*f0²-(k/(7²*f0))*f0+x/f(7); =>-cy^2-dy+x/f(7) (c=k%7/f(7),d=k/f(49));1)

* n=>x/(7f0)-(k/7²)-f0*(k%7)/7;2)

{证明法1)::if e(n)=x; then d(x)=n,假定正确,证明如下:

由R:e(n)得,

x=e(n)=ay^2+by; (a=k%7,b=7n+k/7,y=f0);

由R:d(x)得

n=d(x)=-cy^2-dy+x/f(7); (c=k%7/f(7),d=k/f(49));

代入x,a,b

=>n=d(x)=>-cy^2-dy+((k%7)f0²+(7n+k/7)*f0)/f(7);

代入c,d

=>n=-(k%7/f(7))y^2-(k/f(49))y+((k%7)y²+(7n+k/7)*y)/f(7);

=>n=-(k%7/f(7))y^2-(k/f(49))y+((k%7)/f(7))y²+((7n+k/7)/f(7))*y;

合并y^2,y

=>n=(-(k%7/f(7))+((k%7)/f(7)))y^2+(-(k/f(49))+(7n+k/7)/f(7))y;

=>n=(0)y2+(-(k/f(49))+(7n+k/7)/f(7))y;

=>n=(-k/f(49)+7n/f(7)+k/f(49))y;

=>n=(7n/f(7))y;

=>n=(7n/7*f0)y;

=>n=(n/f0)y;

as y=f(0)=f0;

=>n=(n/f0)*f0;

=>n=n 成立;

证毕.

}

R1:x*f(n)=n*f(x);

R2:d(x)=n,e(n)=x;

*

*{证明法2: if e(n)=x; then d(x)=n; 假定正确,证明如下:

*令

* R1) x=e(n)=(k%7)f0²+(7n+k/7)*f0;

* R2) n=d(x)=x/(7f0)-(k/7²)-f0*(k%7)/7;

*

* 由R2:d(x)

=>n=d(x)=x/(7f0)-(k/7²)-f0(k%7)/7;3)

=>x=(n+k/7²+(k%7)*f0/7)*(7f0);

*

* =>x=7n*f0+k*f0/7+(k%7)*f0²;

由R1:x=e(n)=(k%7)f0²+(7n+k/7)*f0;

* 7n*f0+k*f0/7+(k%7)*f0²=(k%7)f0²+(7n+k/7)*f0;

=>7n*f0+k*f0/7=7n*f0+f0*k/7;

=>k*f0/7=f0*k/7;

=>k*f0=f0*k 乘法交换律成立;

证毕.

* }

2.2.2 基于2.2.1

通过R2:3)

知:n=d(x)=x/(7f0)-(k/7²)-f0*(k%7)/7;x是密文,k是密钥

k,n 均不为0的情况下,每个二元一次方程都有无数对方程的解,导致求解困难,反证加密算法安全性.

2.2.3 基于2.2.1

通过R1:1)

知:x=ay²+by=>ay²+by-x=0如果x,y互换=>y=ax²+bx; (a=k%7,b=7n+k/7,x=f0,a!=0);

由y=ax²+bx ,a=k%7,b=7n+k/7,x=f0,由公式知,当k值变化一点,y值变化很大;

这样很小的k的变化会引起很大的加密数据的改变,y相对于n值变化就越大,信息失真度就越明细,原始信息保密性就越强大。

[0007] 3基于2信息加密、解密算法结合Base84/60编码(查同期专利)可实现网站令牌的高倍高效压缩加密技术方案,其特征如下:

3.1 用户数据令s=user.p.toStr(),s为以&…= 为格式的用户基础数据,其来源于上次登录成功后数据调用之AFS(可迭代分布式存储引擎),明文数据长=1613字节;基于本发明3编码Base84编码再进行加密,令e=MagicNumberEncrypt.encrypt(k,s);得到密文e;e密文长2361字节,耗时:10812751ns,解密:9239864ns;也即加密压缩比146%,远小于传统的200%压缩比;AES:压缩后密文长度:2200字节,压缩比:136%,AES:耗时:714830523ns,AES解密:2755762ns,压缩比对比:146:136,加密速度:72.3:1,解密速度:0.27:1,安全性:前者比后者更强;

3.2 基于2对信息加密过程,需要进行幻数计算,因幻数存储空间较大,利用密文被幻数整除原理需要对密文进行压缩处理,z函数由此诞生,产生压缩码对压缩码进行base84/60编码,再合并发送;

3.3 基于2,3.2对信息解密过程MagicNumberEncrypt.decrypt(k,s),先对信息进行Base84/60解码,再合并,再进行逆还原通过u函数对压缩码进行解码还原为大幻数,再对幻数进行d(k,s)解密。

[0008] 4基于3可实现网站令牌的无登录认证技术:用户基本信息被加密,存储在cookie,user键,总长约2.3kb,不超过4K,用户端无密钥,且密钥不传输,仅在服务端进行解密令牌,通过OTP_SN口令可实现将流密钥动态刷写于令牌,通过浏览器及时交互往返传输于用户端与服务器端,OTP_SN1分钟自动过期,其密钥可通过IPSC/IP技术(查前专利)通过js无痕代码经过计算求得OTP_SN,js端拿到OTP_SN可实现对url参数进行客户端sign签名提交数据,以保护数据安全,其表现为当登录端,输入账户,密码,鼠标离开密码输入框失去焦点时,密码,自动调整为较初始值更长的密文;

OTP_SN时效过期特征:表现在当在提交订单支付前,服务端提示:“签名无效,返回刷新重新提交”类提示,此时因为web端OTP_SN已经过期,返回前一步重新刷新再提交即可通过,因为服务端Msession.genOTP_TLSP算法会自动产生基于时间不可预测LSP公平算法(查前专利)密钥发生器,而系统在解密token令牌前会自动读取最新的密钥SN并加密推入令牌并2次加密。

[0009] 5基于4可实现在https/http协议非安全协议实现轻量级较为可靠的安全保密通信,基于4可实现签名防篡改技术。

[0010] 6基于1,2幻数定理,可广泛应用于军用,民用,航空,航天大数超算,以及信息安全领域,其表现为较快的运算速度,较快的信息加密速度,以及较高的信息安全破解难度。

[0011] 7基于4,5可广泛应用于离线超算与离线存储,离线交易结算,离线钱包功能,具体表现所为离线是指常时间的不操作AFS存储器,所有的计算,与交易结算可缓存在cookie端加密存储,可在多个分布微交易后一次性push到AFS存储器进行数据同步更新操作,这样可减少数据存储的频繁交互,减轻服务端压力。

[0012] 8基于1幻数,其思路类比珠算,可改变传统10进制乘法,通过幻数的高低位移位运算可代替传统的10进制乘法运算,使得运算效率提示60倍。

[0013] 现实意义:

SMG-wscomm-MagicNumberEncrypt一种基于幻数定理高效高压缩数据加密技术:

本发明为基于幻数定理推导出一系列公式,可实现对任意自然数通过与基幻数发生作用产生新的幻数,通过幻数运算将自然数乘法改变为幻数以10为底进行高低位移位加法运算,直接提高运算效率几个数量级,并通过一系列规则公式推导,实现信息的安全加密、解密;较传统AES加密效率提升60倍,本发明为颠覆性技术超越传统计算理论,甚至改变数学乘法基础;本发明主要用于动态令牌加密编码传输;以及航天航空领域的大量观测大数进行压缩,结合Base84算法(参同期专利)可实现高安全性,高可靠性信息保密压缩传输;高倍空间压缩编码,高效率加密解密技术为超算力输出,超分布式存储输出必放一异彩。

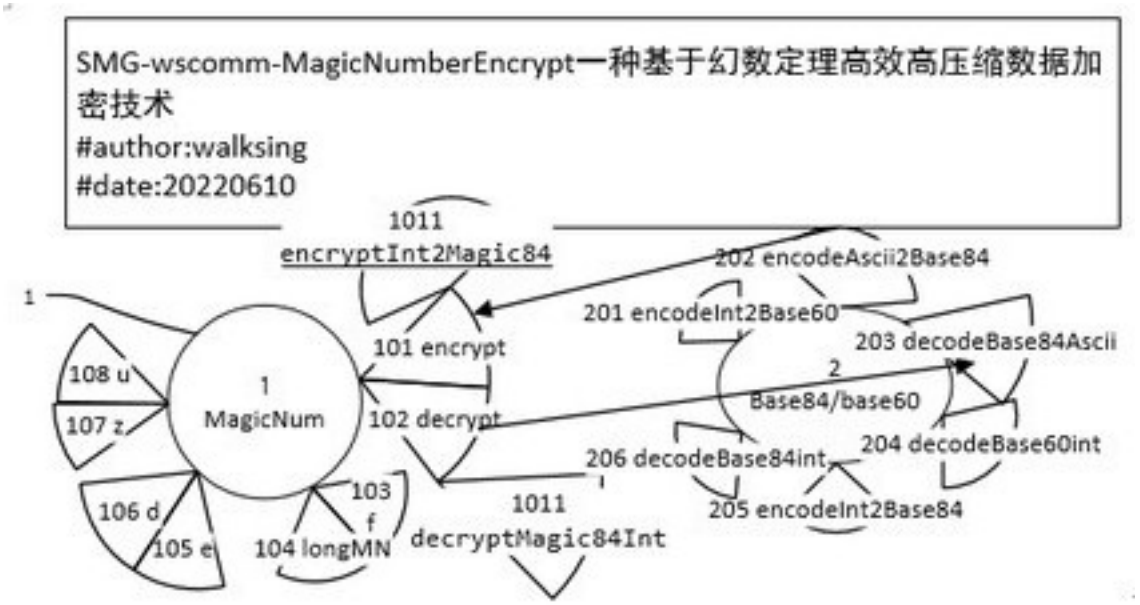

SMG-wscomm-MagicNumberEncrypt一种基于幻数定理高效高压缩数据加密技术
#author:walksing
#date:20220610
1011
encryptInt2Magic84
202 encodeAscii2Base84
201 encodeInt2Base60
203 decodeBase84Ascii
101 encrypt
102 decrypt
1
MagicNum
108 u
107 z
106 d
105 e
104 longMN
103
f
1011
decryptMagic84Int
206 decodeBase84int
205 encodeInt2Base84
204 decodeBase60int
2
Base84/base60

图1

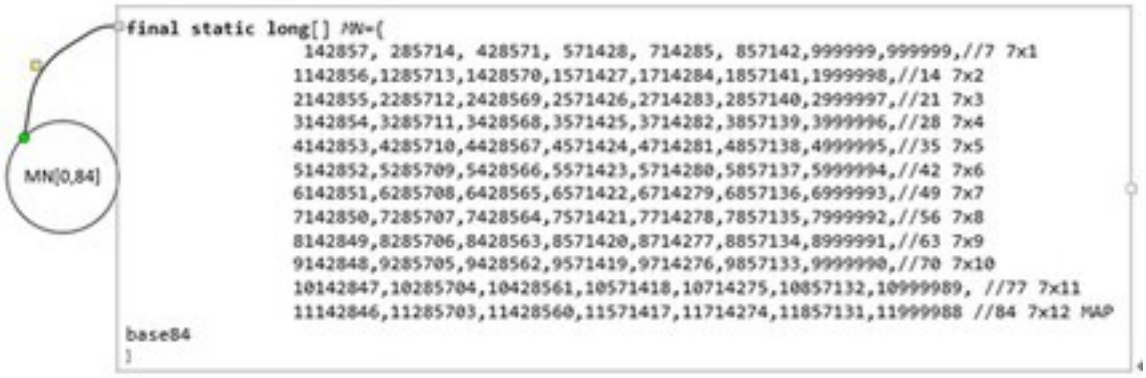

final static long[] MN={
142857, 285714, 428571, 571428, 714285, 857142,999999,999999,//7 7x1
1142856,1285713,1428570,1571427,1714284,1857141,1999998,//14 7x2
2142855,2285712,2428569,2571426,2714283,2857140,2999997,//21 7x3
3142854,3285711,3428568,3571425,3714282,3857139,3999996,//28 7x4
4142853,4285710,4428567,4571424,4714281,4857138,4999995,//35 7x5
5142852,5285709,5428566,5571423,5714280,5857137,5999994,//42 7x6
6142851,6285708,6428565,6571422,6714279,6857136,6999993,//49 7x7
7142850,7285707,7428564,7571421,7714278,7857135,7999992,//56 7x8
8142849,8285706,8428563,8571420,8714277,8857134,8999991,//63 7x9
9142848,9285705,9428562,9571419,9714276,9857133,9999990,//70 7x10
10142847,10285704,10428561,10571418,10714275,10857132,10999989, //77 7x11
11142846,11285703,11428560,11571417,11714274,11857131,11999988 //84 7x12 MAP
base84
}
MN[0,84]

图2

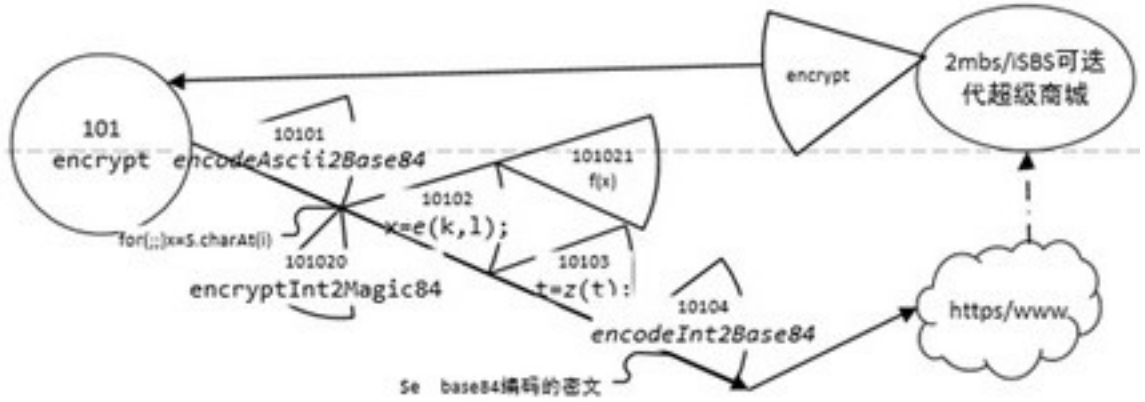

101
encrypt
encodeAscii2Base84
10101
101021
f(x)
10102
x=e(k,1);
for(;;)x=S.charAt(i)
101020
encryptInt2Magic84
10103
t=z(t);
10104
encodeInt2Base84
Se base84编码的密文
encrypt
2mbs/ISBS可选
代超级商城
https/www.

图3

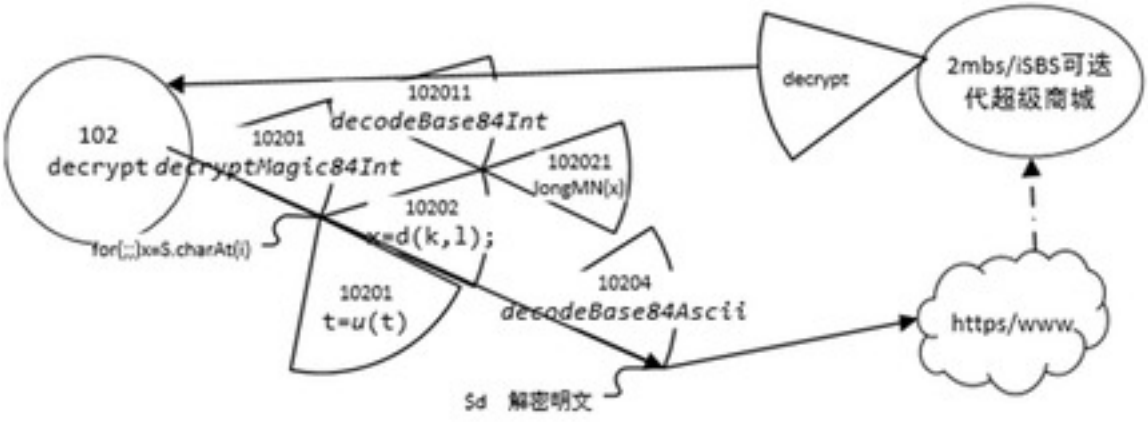

decrypt
2mbs/iSBS可选
代超级商城
102011
decodeBase84Int
102
decrypt decryptMagic84Int
10201
102021
longMN(x)
10202
for(;;)x=S.charAt(i)
10201
t=u(t)
10204
decodeBase84Ascii
https/www.
Sd 解密明文

图4