基于随机背包公钥密码的攻击*

古春生[1,2]，于志敏[1]，景征骏[1,3]
(1. 江苏技术师范学院 计算机工程学院，江苏 常州 213001；2. 中国科学技术大学 计算机科学与技术学院，合肥 230027；3. 南京邮电大学 计算机学院，南京 210003)

摘 要：针对基于随机背包公钥密码方案，根据方案不同参数分别给出了恢复私钥攻击和恢复密文中明文的格攻击，并通过计算实例验证格攻击有效。因此证明了基于随机背包中公钥方案是不安全的。

关键词：公钥密码体制；陷门背包；密码分析；格攻击
中图分类号：TP309 **文献标志码**：A **文章编号**：1001-3695(2012)09-3486-03
doi:10.3969/j.issn.1001-3695.2012.09.076

Attack on random knapsack-based public key cryptosystems

GU Chun-sheng[1,2], YU Zhi-min[1], JING Zheng-jun[1,3]
(1. *School of Computer Engineering, Jiangsu Teachers University of Technology, Changzhou Jiangsu* 213001, *China*; 2. *School of Computer Science & Technology, University of Science & Technology of China, Hefei* 230027, *China*; 3. *School of Computer Science, Nanjing University of Posts & Telecommunications, Nanjing* 210003, *China*)

Abstract: For the random knapsack-based public key cryptosystems, this paper respectively presented an attack of recovering secret key and lattice attack of recovering plaintext from ciphertext according to the different parameter settings. The computing experiments verify the effectiveness of the lattice attack. Hence, the random knapsack-based public key cryptosystems is not secure.

Key words: public-key cryptosystem; trapdoor knapsack; cryptanalysis; lattice attack

0 引言

自从 Merkle 等人[1]提出第一个基于背包问题的公钥密码方案以来，研究人员设计出许多基于背包问题的密码方案。该类方案易于受到 Shamir 攻击[2]、低密度攻击[3]和联立丢番图逼近攻击[4]，设计抵抗这些攻击的新型背包公钥密码方案一直是公钥密码研究的热点之一。对于分析证明能够抗 Shamir 攻击和低密度攻击的背包公钥方案[5~9]，通常还存在其他有效攻击方法[10~14]。

文献[15]构造了新的基于随机背包问题的高效公钥加密算法，该公钥算法加解密时计算简单，运算速度快。文献[15]证明在攻击者不掌握私钥信息情况下该密码算法能够抵抗直接求解背包问题攻击，包括低密度攻击和联立丢番图逼近攻击等。

本文主要研究分析文献[15]中基于随机背包公钥密码方案的安全性。针对不同参数环境，分别给出从公钥直接恢复私钥攻击和从密文直接恢复明文的格攻击。针对背包公钥方案，目前的格攻击方法[2~4]基本上都使用归约基中最短向量作为背包向量。实际上，在高密度背包情况下，背包向量不一定对应于格的最短向量，且使用 LLL 算法也不一定能求出格的最短向量。因此，针对高密度背包(背包密度大于 0.9408)，本文使用文献[16]中格攻击方法变种，即直接调用 LLL 算法求解归约基，在归约基中寻找与明文背包向量对应的特殊向量。

1 攻击基于随机背包的公钥密码

1.1 基于随机背包的公钥密码[15]

1)密钥生成算法 随机选取 n 维背包向量 $\boldsymbol{U}=(u_1,\cdots,u_n)$，$u_i(i=1,\cdots,n)$为正整数，向量 $\boldsymbol{V}=(v_1,\cdots,v_n)$满足关系 $v_i=u_i-2^{n-i}$。随机选取两个不同的素数 p、q 满足关系

$$p>\sum_{i=1}^{n}u_i,\ q>2\max\{\sum_{v_i>0}v_i-\sum_{v_i<0}v_i\} \tag{1}$$

使用中国剩余定理计算向量 $\boldsymbol{A}=(a_1,\cdots,a_n)(0<a_i<pq)$ 满足关系：

$$a_i=u_i \bmod p,\ a_i=v_i \bmod q \quad i=1,\cdots,n \tag{2}$$

输出背包向量 $\boldsymbol{A}=(a_1,\cdots,a_n)$为公钥，$p$、$q$ 为私钥。

2)加密算法 给定公钥向量 $\boldsymbol{A}$ 和 n bit 长的二进制明文 $(m)_2=m_1\cdots m_n$，输出密文 $c=\sum_{i=1}^{n}m_ia_i$。

3)解密算法 给定私钥 p、q 和密文 c，计算 $c_p=c \bmod p$ 和 $c_q=c \bmod q$，其中 c_p 取模 p 的非负最小剩余，即 $0\leqslant c_p<p$；c_q 取模 q 的绝对最小剩余，即 $-q/2<c_q<q/2$。输出明文 $(m)_2=(c_p-c_q)_2$。

1.2 攻击方法

文献[15]的背包公钥密码方案在密钥生成算法中，并未明确在什么范围内随机选取向量 $\boldsymbol{U}$，选择的素数 p、q 也存在这个问题。为分析该方案安全性，不失一般性地假定 n 为安全参数，它也是明文比特长度。为表述方便，对于任意向量 $\boldsymbol{U}$，记

收稿日期：2012-02-29；**修回日期**：2012-03-26 **基金项目**：国家自然科学基金资助项目(61142007)；江苏省“青蓝工程”基金资助项目(KYQ900Z)；江苏技术师范学院基金资助项目(KYY110055)

作者简介：古春生(1971-)，男，副教授，博士，主要研究方向为公钥密码分析(guchunsheng@gmail.com)；于志敏(1974-)，男，讲师，硕士；景征骏(1974-)，男，讲师，博士研究生.

$\|U\|_\infty=\max\{|u_i||u_i\in U\}$,$|k|$为整数$k$的比特长度。

由信息论知公钥向量A满足$|\|A\|_\infty|\geq n$。本文攻击方法根据公钥参数$|\|A\|_\infty|$不同分三种情况:a)$|\|A\|_\infty|=n+O(\log n)$;b)$|\|A\|_\infty|=n+O(n^\varepsilon)$;c)$|\|A\|_\infty|\geq 2n+O(\log n)$。下面针对不同参数使用不同攻击方法。

定理 1　如果向量A的最大范数比特长度为$|\|A\|_\infty|=n+O(\log n)$,则存在多项式时间算法直接恢复私钥。

证明　因为公钥$A=(a_1,\cdots,a_n)$的比特长度为$|\|A\|_\infty|=n+O(\log n)$,故$N$比特长度为$|N|=n+O(\log n)$。又因$u_i$为正整数,且$v_i=u_i-2^{n-i}(i=1,\cdots,n)$,则除非随机选择的向量$U$基本上是一递增向量(这种情况发生概率可以忽略不计),否则以高概率$\|V\|_\infty=O(2^n)$。故不失一般性地设$|\|V\|_\infty|=n+O(\log n)$。由$q>2\max\{\sum_{v_i>0}v_i-\sum_{v_i<0}v_i\}$知$|q|=n+O(\log n)$。又由$N=pq$和$|N|=n+O(\log n)$可以推得$|p|=O(\log n)$。再由$p>\sum_{i=1}^n u_i$和$u_i$为正整数得到$\|U\|_\infty=n^{O(1)}$。

根据关系$a_i=u_i \bmod p$,枚举猜测u_1、u_2(实际上可以随机猜测任意两个元素),使用欧几里德算法计算最大公因式$\gcd(a_1-u_1,a_2-u_2)$。因为$\|U\|_\infty=n^{O(1)}$,故可能的u_1、u_2组合只有多项式个,因此多项式时间内一定可以正确得到u_1、u_2,并计算出私钥p,然后利用关系$v_i=u_i-2^{n-i}$和$a_i=v_i \bmod q$易于求得私钥q。

同时,易于验证如果小概率事件发生,即随机选择的向量U基本上是一个递增向量,且$\|V\|_\infty=n^{O(1)}$,则上述方法计算得到私钥q。此时利用关系$a_i=v_i \bmod q$和$v_i=u_i-2^{n-i}$,也易于求解$u_i(i=1,\cdots,n)$和私钥p。

算法复杂性分析。判断整数k的比特长度需时间$O(\log k)$。公钥向量A中有n个整数,每个整数长度至多为$n+O(\log n)$,故此步计算需要的时间为$O(n\times(n+\log n))=O(n^2)$。由上述推理假设$\|U\|_\infty=n^{O(1)}$。因此计算私钥$p$需要猜测$n^{O(1)}\times n^{O(1)}=n^{O(1)}$对可能的$u_1$、$u_2$。对于每对$u_1$、$u_2$,计算$\gcd(a_1-u_1,a_2-u_2)$需要的时间至多为$O(n^3)$。故计算私钥$p$需要的时间为$n^{O(1)}\times O(n^3)=n^{O(1)}$。在计算出私钥$p$以后,计算私钥$q$仅需要计算$a_j=u_j \bmod p$,$v_j=u_j-2^{n-j}(j=1,2)$和$q=\gcd(a_1-v_1,a_2-v_2)$,这步需要的时间至多为$O(n^3)$。因此,如果$|\|A\|_\infty|=n+O(\log n)$,则在多项式时间内可以直接恢复私钥。

定理 2　如果向量A最大范数比特长度为$|\|A\|_\infty|=2n+O(\log n)$,则存在多项式时间算法以不可忽略概率直接输出明文。

证明　因为$|\|A\|_\infty|=2n+O(\log n)$,容易计算出该方案背包密度约为$n/(2n+O(\log n))\approx 0.5$。Coster 等人[3]证明若背包密度小于 0.9408,则使用格归约算法能以不可忽略概率在多项式时间内求解密文中明文。

评论 1　尽管文献[15]中未明确设置参数大小,但随机产生的公钥向量符合这种情况的可能性最大。另外,在文献[15]中为对密钥信息进一步隐藏,给出了改进密钥生成算法。随机选择$U=(u_1,\cdots,u_n)$、$V=(v_1,\cdots,v_n)$,满足$v_i=u_i-2^{n-i}$和一个二阶整数可逆矩阵W,记其逆矩阵为W^{-1}。计算向量$(G,H)^{\mathrm{T}}=W\times(U,V)^{\mathrm{T}}$。类似前面密钥生成算法,根据向量$G$、$H$选择随机素数$p$、$q$,并计算输出公钥$pk=(A)$和私钥$sk=(p,q,W)$,加密算法不变。解密计算时,$c_p$、$c_q$取模$p$、$q$的绝对最小剩余,并计算$(s_p,s_q)^{\mathrm{T}}=W^{-1}\times(c_p,c_q)^{\mathrm{T}}$和输出明文$(m)_2=(s_p-s_q)_2$。易于验证$|\|G\|_\infty|\approx|\|H\|_\infty|\approx n$,即向量$G$、$H$最大范数比特长度大致相等。此时方案背包密度仍然符合定理 2 中格归约攻击情况。

从定理 1、2 可以看出,$|\|U\|_\infty|$不能太小,也不能太大。因此假定$|\|U\|_\infty|=O(n^\varepsilon)$、$|\|A\|_\infty|=n+O(n^\varepsilon)$,方案背包密度为$n/(n+O(n^\varepsilon))>0.9408$。对于这种高密度背包参数,本文使用文献[16]中格攻击方法的变种,设计启发式格攻击方法。尽管未能从理论上分析该启发式格攻击方法性能,但从计算实验结果看,该方法非常有效(在所有计算实验中100%成功),从而计算证明了在高密度参数情况下公钥方案[15]也不安全。

启发式格攻击算法设计如下:

a)给定公钥$A=(a_1,\cdots,a_n)$和密文$c=\sum_{i=1}^n m_i a_i$,构造格L如下:

$$L=\begin{pmatrix} c & -1 & -1 & \cdots & -1 & 1\\ a_1 & 2 & 0 & \cdots & 0 & 0\\ a_2 & 0 & 2 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & & \vdots & \vdots\\ a_n & 0 & 0 & \cdots & 2 & 0 \end{pmatrix}$$

b)调用 LLL 算法[17],得到格L的归约基B。

c)在基B中寻找如下形式的向量$B_j=(B_{j,0},B_{j,1},B_{j,2},\cdots,B_{j,n},B_{j,n+1})$,满足条件$B_{j,0}=0$,$B_{j,n+1}=1$,$B_{j,i}=1$或$B_{j,i}=-1(i=1,\cdots,n)$。

d)如果在基B中找到这样一个向量,则输出明文$m=(m_1,\cdots,m_n)$,其构造如下:

$$\begin{cases} m_i=1 & \text{如果 } B_{j,i}=1\\ m_i=0 & \text{如果 } B_{j,i}=-1 \end{cases}(i=1,\cdots,n)$$

分析启发式格攻击算法的算法复杂性:调用 LLL 算法需要的时间为$n^6\log^3(n\|A\|_\infty)\approx n^6\log^3 n$(实际时间远小于这个时间界),扫描查找特定向量$B_j$的时间为$O(n^2)$。故该启发式格攻击需花费多项式时间。

1.3　格攻击计算实验

本文使用 NTL 库[18]验证上述格攻击算法的有效性。为节省空间,下面仅给出$n=1024$的具体参数值:

$|p|=12$,$|q|=1025$,$|N|=|pq|=1037$,背包密度为$1024/1037\approx 0.9875$。

$p=3061$

$q=$
359538626972463181545861038157804946723595395788461314
546860162315465351611001926265416954644815072042240277597427
867153175795376288332449856948612789482487555357868497309705 5
260443920249218823890616590417001153767630136468492576294782 6
221081654474326701021369172596479894491876959432609670712659 2
48448268389

$U=$[4 3 3 2 1 2 4 1 4 1 5 4 4 2 2 2 2 4 5 5 3 3 1 5 3 5 2 5 5 5 4 5
2 4 5 5 1 3 5 3 4 5 3 2 5 2 4 1 1 2 5 2 3 2 4 3 3 4 2 3 1 5 5 5 5 1 1 3 1 4
2 5 4 1 4 1 2 4 3 1 4 3 4 1 5 2 2 2 4 3 3 1 4 1 3 2 4 1 4 1 5 3 4 5 5 3 2 5
2 1 3 5 1 1 5 2 1 5 1 4 1 1 5 4 4 5 4 5 5 4 1 5 3 4 2 4 5 1 5 2 5 1 4 2 1 3
4 1 2 5 5 2 5 2 5 1 3 4 5 2 5 2 3 3 2 5 4 1 4 4 3 1 3 3 3 4 2 3 5 1 5 1 2 4
4 4 2 1 5 2 2 5 2 4 1 5 4 2 4 2 5 2 3 3 1 2 4 2 4 1 3 1 3 3 2 1 1 3 3 1 5 2
2 3 4 1 3 1 5 3 1 3 1 4 5 3 1 5 4 4 4 5 3 4 2 4 2 3 1 5 3 2 3 1 1 4 2 4 5 4
4 4 5 1 1 2 5 3 5 1 4 2 5 1 2 3 1 1 5 5 2 3 2 4 1 4 3 3 5 2 4 5 5 3 1 4 1 1
1 5 4 5 3 5 1 3 3 4 4 1 4 1 1 2 2 3 5 4 3 5 4 5 2 4 2 2 2 1 1 4 1 2 5 1 4 1
1 2 3 4 2 4 4 2 1 1 4 4 2 1 1 2 4 2 2 1 2 1 2 1 2 2 1 2 1 3 2 3 3 4 4 1 4 3
1 3 4 5 2 1 5 3 1 1 1 5 1 3 3 4 3 3 4 4 2 5 5 5 2 2 4 4 3 4 3 2 5 3 4 4 5 5
4 4 5 1 1 1 2 5 4 5 3 5 5 3 1 3 3 5 3 5 4 1 2 3 5 1 2 3 2 4 5 2 4 2 2 3 1 4

1 3 3 4 1 1 1 5 1 2 3 1 4 3 4 2 3 2 5 2 2 1 1 2 2 5 5 3 4 1 2 2 1 4 5 2 3 2
5 5 5 1 4 2 1 4 1 3 5 1 4 5 1 4 2 5 2 1 1 4 2 2 4 5 5 3 2 5 2 4 3 1 4 4 3 1
2 4 2 2 1 2 3 1 5 1 4 2 1 1 4 2 1 1 2 2 3 1 5 4 3 3 4 3 3 1 1 1 5 2 2 4 4 4
4 5 4 5 4 2 3 5 3 1 5 1 2 5 2 3 5 1 4 3 2 5 4 3 5 1 2 1 4 3 4 4 4 5 5 3 1 2
3 2 2 4 3 3 5 2 1 3 2 2 2 5 4 5 2 1 1 1 1 5 1 4 4 2 4 3 1 3 2 1 1 5 4 1 2 2
4 5 4 1 1 3 5 5 4 5 2 2 5 1 1 5 5 3 1 3 3 4 4 4 1 1 3 3 5 3 4 4 5 4 2 1 3 1
3 3 4 3 5 1 4 3 2 2 2 3 5 4 2 5 3 1 2 3 3 5 3 2 5 1 4 5 1 3 2 5 2 3 3 3 5 5
5 4 5 5 3 2 1 2 4 3 4 4 1 5 2 3 2 1 1 4 2 3 4 1 4 5 3 3 1 1 4 5 1 3 2 4 2 4
3 2 3 5 2 5 3 2 3 2 5 1 4 4 5 3 5 4 4 3 5 2 5 4 4 3 4 1 4 2 4 5 1 5 3 5 4 5
2 2 2 5 5 1 5 3 1 4 3 1 1 4 2 3 5 1 3 4 5 1 2 4 1 2 5 5 2 3 1 1 5 1 3 4 1 1
3 1 5 3 5 4 3 3 5 5 1 1 4 2 1 5 3 1 1 4 2 4 4 5 3 2 4 3 4 2 3 5 4 2 4 1 3 1
4 1 5 4 3 4 2 4 4 1 1 3 4 2 2 1 1 1 4 2 2 3 3 3 1 4 4 3 2 4 4 2 3 1 1 3 3 1
3 2 1 2 3 4 2 3 3 4 3 5 2 4 5 4 1 4 1 4 1 2 3 3 5 4 1 1 3 4 3 1 4 2 3 4 2 1
4 4 3 1 4 1 5 1 4 5 4 3 5 3 4 4 4 2 2 2 4 5 2 1 5 5 4 5 5 3 4 1 5 2 3 5 3 4
5 1 4 3 2 2 1 2 4 3 5 3 1 4 5 3 5 3 2 4 4 1 3 5 3 4 5 1 3 1 3 1 5 2 5 2 4 3
2 4 2 3]

根据上述参数使用密钥生成算法可以计算出方案公钥 $A=(a_1,\cdots,a_n)$，然后使用加密算法对任意二进制明文背包向量 m 进行加密，输出密文 $c=\sum_{i=1}^{n}m_ia_i$，最后直接使用上述启发式格攻击算法可以恢复明文 m。

在上述实验中，向量 U 的比特长度非常小，目的是为了验证本文提出的格攻击方法在高密度背包下的有效性。计算实验结果表明仅依靠提高文献[15]中公钥方案的背包密度无法抵抗上述启发式格攻击算法。

针对文献[15]的不同参数值和背包密度，计算实验结果（如表 1 所示）表明本文格攻击算法有效。因此从计算实验角度看，文献[15]中公钥方案是不安全的。

表 1 格攻击实验结果

n	$\lvert\lVert U\rVert_\infty\rvert$	$\lvert\lVert V\rVert_\infty\rvert$	$\lvert p\rvert$	$\lvert q\rvert$	$\lvert N\rvert$	$\lvert\lVert A\rVert_\infty\rvert$	背包密度 $n/\lvert N\rvert$	实验次数	成功次数
300	10	300	18	301	319	319	0.940 4	10	10
400	10	400	18	401	419	419	0.954 7	10	10
500	10	500	18	501	519	519	0.963 4	10	10
600	10	600	19	601	620	620	0.967 7	10	10
700	3	700	11	701	712	712	0.983 1	10	10
800	3	800	11	801	812	812	0.985 2	10	10
1 024	3	1 024	12	1 025	1 037	1 037	0.987 5	10	10

2 结束语

本文研究分析了文献[15]中基于随机背包的公钥方案安全性。针对文献[15]中方案的不同参数，本文分别给出了不同多项式时间的攻击方法。针对在高密度背包参数的公钥方案，通过计算实验验证了格攻击方法的有效性。

参考文献：

[1] MERKLE R C, HELLMAN M E. Hiding information and signatures in trapdoor knapsacks[J]. IEEE Trans on Information Theory, 1978,24(5):525-530.

[2] SHAMIR A. A polynomial-time algorithm for breaking the basic Merkle-Hellman cryptosystem[J]. IEEE Trans on Information Theory,1984,30(5):699-704.

[3] COSTER M J, JOUX A, LAMACCHIA B A, *et al*. Improved low-density subset sum algorithms[J]. Computational Complexity, 1992,2(2):111-128.

[4] LAGARIAS J C. Knapsack public key cryptosystems and Diophantine approximation[C]//Proc of CRYPTO on Advances in Cryptology. New York:Plenum,1984:3-23.

[5] 王保仓,胡予濮. 高密度背包型公钥密码体制的设计[J]. 电子与信息学报,2006,28(12):2390-2393.

[6] WANG Bao-cang, WU Qian-hong, HU Yu-pu. A knapsack-based probabilistic encryption scheme[J]. Information Sciences, 2007, 177(19):3981-3994.

[7] 张卫东,王保仓,胡予濮. 一种新的背包型公钥密码算法[J]. 西安电子科技大学学报,2009,36(3):506-511.

[8] WANG Bao-cang, HU Yu-pu. Quadratic compact knapsack public-key cryptosystem[J]. Computers & Mathematics with Applications,2010,59(1):194-206.

[9] ABOUD S J. An improved knapsack public key cryptography system [J]. International Journal of Internet Technology and Secured Trans,2011,3(3):310-319.

[10] 韩立东, 刘明洁, 毕经国. 两种背包的公钥密码算法的安全性分析[J]. 电子与信息学报,2010,32(6):1485-1488.

[11] YOUSSEF A M. Cryptanalysis of a knapsack-based probabilistic encryption scheme[J]. Information Sciences,2009,179(18):3116-3121.

[12] YOUSSEF A M. Cryptanalysis of a quadratic knapsack cryptosystem [J]. Computers & Mathematics with Applications,2001,61(4): 1261-1265.

[13] LEE M S. Cryptanalysis of a quadratic compact knapsack public-key cryptosystem[J]. Computers & Mathematics with Applications, 2011,62(9):3614-3621.

[14] 潘彦丰, 杨卫武. 对一种基于 Euler-Fermat 小定理的背包公钥系统的攻击[J]. 信息工程大学学报,2011,12(5):532-534.

[15] 王保仓, 韦永壮, 胡予濮. 基于随机背包的公钥密码[J]. 电子与信息学报,2010,32(7):1580-1584.

[16] LAGARIAS J C, ODLYZKO A M. Solving low-density subset sum problems[J]. Journal of the ACM,1985,32(1):229-246.

[17] LENSTRA A K, LENSTRA H W, LOVASZ L. Factoring polynomials with rational coefficients[J]. Mathematische Annalen, 1982, 261 (4):515-534.

[18] SHOUP V. NTL:a library for doing number theory[EB/OL]. (2009-08-14). http://shoup.net/ntl/.