Weaknesses of a public-key cryptosystem based on factorizations of finite groups

Simon Blackburn * Sean Murphy
Information Security Group,
Royal Holloway and Bedford New College,
University of London,
Egham, Surrey TW20 0EX, U.K.

Jacques Stern
Laboratoire d'Informatique,
Ecole Normale Supérieure,
45, rue d'Ulm,
75230 Paris

Abstract. Recently, Qu and Vanstone have announced the construction of several new public-key cryptosystems based on group factorization. One of these was described at the last AUSCRYPT meeting [2]. We point out a serious weakness of this last system which makes it insecure. Our method only uses elementary algebra.

1 The proposed cryptosystem

Let G be a finite group. A *factorization* of G is a sequence $A_1, \cdots, A_s$ of subsets of G such that each element g of G can be expressed uniquely as a product

$$g = g_s g_{s-1} \cdots g_1$$

where $g_i \in A_i$.

The public-key cryptosystem described in [2] uses the additive group $G = Z_2^n$. Starting from a sequence $\alpha_1, \cdots, \alpha_n$ of generators, the authors build a sequence $G = G_0 > G_1 > \cdots > G_{n/2} = \{0\}$ of subgroups, where G_i is generated by $\{\alpha_{2i+1}, \alpha_{2i+2}, \alpha_{2i+3}, \cdots, \alpha_n\}$. Next, a complete set of coset representatives of G_i in G_{i-1}

$$\overline{A}_i = \{\overline{\alpha}[i,0], \overline{\alpha}[i,1], \overline{\alpha}[i,2], \overline{\alpha}[i,3]\}$$

is chosen, where

$$\overline{\alpha}[i,j] = j_1 \alpha_{2i-1} + j_2 \alpha_{2i} + \alpha_{i,j}$$

$j_1 j_2$ being the binary expansion of j and $\alpha_{i,j}$ a random element of G_i.

The family $\overline{A}_i$ is defined for $i \leq n/4 - 1$. For $n/4 \leq i \leq n/2 - 2$, a somehow similar construction is performed, with the difference that four complete sets

$$\overline{A}_{i,h} = \{\overline{\alpha}[i,0]_h, \overline{\alpha}[i,1]_h, \overline{\alpha}[i,2]_h, \overline{\alpha}[i,3]_h\}$$

are built instead of one.

* This author was supported by S.E.R.C. research grant GR/H23719

T. Helleseth (Ed.): Advances in Cryptology - EUROCRYPT '93, LNCS 765, pp. 50-54, 1994.

Finally, a one-one function f that maps the set $\{1, \cdots, n/4-1\}$ onto the set $\{n/4, \cdots, n/2-2\}$ is chosen and the public key $A_1, \cdots, A_{n/4-1}$ is defined where A_i is obtained by (randomly) ordering the set

$$\bigcup_{\substack{0 \le j \le 3 \\ 0 \le h \le 3}} \{\overline{\alpha}[i,h] + \overline{\alpha}[f(i),j]_h\}$$

as

$$\alpha[i,p] \qquad , \qquad 0 \le p \le 15$$

A message m of $n-4$ bits is encoded as follows: packing the bits 4 by 4, one obtains a sequence $m_1, \cdots, m_{n/4-1}$ of hexadecimal digits from which the ciphertext c is computed as:

$$c = \sum \alpha[i, m_i]$$

The secret data coming from the construction itself allow decoding: using the decomposition of c in the basis $\alpha_1, \cdots, \alpha_n$ successively gives, for each $i = 1 \cdots n/4-1$, the corresponding index h of the vector

$$\overline{\alpha}[i,h] + \overline{\alpha}[f(i),j]_h$$

chosen from A_i via m_i. But, once all the values h are known, the decomposition of c in the basis $\alpha_1, \cdots, \alpha_n$ also gives by an easy recursion on $k = n/4 \cdots n/2-2$ the missing part $\overline{\alpha}[k,j]_h$ of the vector

$$\overline{\alpha}[f^{-1}(k),h] + \overline{\alpha}[k,j]_h$$

chosen at level $f^{-1}(k)$. All this can be made quite efficient (see [2]). The value $n = 128$ is suggested for practical implementations.

2 Basic observations

In [2], it is stated that, although the cryptosystem is a kind of (modular) knapsack, methods using lattice reduction, such as the Lagarias-Odlyzko attack ([3]) do not apply. We agree with this opinion and we think it remains true even with recent improved versions of this attack such as [1]. Thus, we take another way.

Note that the group G used in section 1 is also a vector space: using Gaussian elimination over the field $Z/2Z$, it is easy to compute the dimension of a subspace of G generated by a given family X of vectors and to output a basis of this subspace.

Also, whenever a large family X is chosen in a subspace F of G, this family is quite likely to generate the entire subspace. For example, we have:

Theorem 1 *If P vectors are chosen independantly at random in a vector space of dimension K over the two element field, the probability that they do not generate the entire space is at most 2^{P-K}.*

The proof of this result is quite simple: given a subspace of dimension $k-1$, the probability that all choices remain in this subspace is at most $\frac{1}{2}^P$. But the number of such subspaces is exactly the number of non-zero linear functionals, i.e. $2^K - 1$.

From the public key of the above cryptosystem we define

$$A^i = \bigcup_{j>i} A_i$$

and we observe the following

Fact: *With high probability, A^i generates the subspace G_i, provided i is not too large.*

Note that A^i contains $16(n/4 - i)$ vectors from the space G_i, which is of dimension $n-2i$. Even if the vectors from A^i are not really chosen independantly, the above theorem still gives a convincing estimate of the probability that A^i does not generate G_i, namely 2^{3n-14i}. For $n = 128$, this estimate remains below 2^{40} up to $i = 27$. This leaves out only four values.

Thus, it is fairly clear that some secret information leeks out. In the next section, we will see how to take advantage of this fact.

3 Cryptanalysis of the system

From section 2, we know that we can recover from the public data the sequence of subgroups G_i, for i not too large, say $i \leq i_0$. Our cryptanalysis include several steps.

Grouping the elements of A_i together, for i not too large. Although the elements of each A_i have been scrambled, it is possible to group together the elements

$$\overline{\alpha}[i,h] + \overline{\alpha}[f(i),j]_h$$

with the same h by using the equivalence relation

$$u \oplus v \in G_i$$

Since, the G_i's are known up to $i = i_0$, the grouping is properly recovered up to $i = i_0$ as well. We note that actually, whenever u and v are equivalent elements of A_i, the sum $u \oplus v$ belongs to $G_{n/4-1}$. This way, we can collect a fairly large family Y of elements of $G_{n/4-1}$.

Extending the method to the last few indices. For i between i_0 and $n/4-1$, we have not been able so far to compute accurately G_i because the sample A^i of elements of G_i was not large enough. Now, if we add to A^i the set Y that has been computed at the end of the last paragraph, then we see that we obtain a generating family for G_i. From this, we can also perform the correct grouping of A_i.

Recovering the secret permutation f. We work again with the equivalence relation on A_i defined above and, this time, we use the fact that, whenever u and v are equivalent elements of A_i, the sum $u \oplus v$ belongs to $G_{f(i)-1}$ (and not only to $G_{n/4-1}$ as was observed above). We let B_i be the set of sums $u \oplus v$ obtained from equivalent elements of A_i. Each B_i contains 24 elements. We define:

$$B^i = \bigcup_{j \neq i} B_i$$

fact:
i) if $f(i) = n/4$ then, with high probability, B^i generates the subspace $G_{n/4}$
ii) otherwise, with high probability, B^i generates the subspace $G_{n/4-1}$

This is because, we have, in each case a very large family of members of the corresponding space.

From the fact, it follows that we can recover both $f^{-1}(n/4)$ and $G_{n/4}$ by computing the dimension of all the spaces generated by the various families B^i. A recursive procedure will then achieve the same for $f^{-1}(n/4+j)$ and $G_{n/4+j}$: this procedure uses the same argument, the family B_i being restricted to those indices i for which $f(i)$ is not yet known (i.e. is $\geq n/4+j$).

Note that, at step j, we have $24(n/4-j-2)$ elements of a subspace of dimension $n/2-2j$ or $n/2-2j+2$. Using the estimate of theorem 1, we see that the probability of error remains quite small even for the last significant case ($j = n/4-3$), for which it is below 2^{-16}. Still, there is a slight chance that $f^{-1}(n/4+j)$ is not correctly computed for say the last two or three values. This issue will be addressed specifically when we turn to decoding.

Decoding. Given the ciphertext c, we first apply the following procedure:
for $i := 1$ to $n/4-1$ do
begin
pick any u in A_i such that $c \oplus u$ belongs to G_i
$u[i] := u$; return $u[i]$;
$c := c \oplus u$;
end;
return c

At the end of the procedure, we have reduced the posible choices of the unique element of A_i that contributes to the sum

$$c = \sum \alpha[i, m_i]$$

to a subset of each A_i consisting of the four elements equivalent to the vector $u[i]$ returned at step i. We denote this subset by $\tilde{A}_i$. Next, we apply the following.
for $i := n/4$ to $n/2-2$ do
begin
pick any v in $\tilde{A}_{f^{-1}(i)}$ such that $c \oplus u[i] \oplus v$ belongs to G_i
$v[i] := v$; return $v[i]$;
$c := c \oplus u$;
end

The value of $v[i]$ returned at step i is the vector $\alpha[i, m_i]$ of the sum

$$c = \sum \alpha[i, m_i]$$

This gives the plaintext m. Efficient implementations using decoding matrices can be implemented as in [2].

As observed above, there is a small chance that a mistake occurs for two or three values $i = f^{-1}(n/4 - 2)$, $i = f^{-1}(n/4 - 3)$, etc. This can be corrected by exhaustive search. Note that, since the mistake comes from the attack (and not from the ciphertext), the proper value of f can be recovered from a few decoding computations.

4 Conclusion

We have pointed out a serious weakness in the system proposed in [2]. Furthermore, we do not feel simple modifications of the system can restore its security. For example, it is quite possible to changer the order of the A_i's but the correct order can be recovered by computing dimensions with the same method we used to disclose f.

References

1. M. J. Coster, A. Joux, B. A. LaMacchia, A. M. Odlyzko, C. P. Schnorr and J. Stern, Improved low-density subset sum algorithms, *Computational Complexity*, to appear.
2. M.Qu, S. A. Vanstone, New public-key cryptosystems based on factorizations of finite groups, *AUSCRYPT'92*, preproceedings page 12.7-12.12.
3. J. C. Lagarias and A. M. Odlyzko, Solving low-density subset sum problems, *J. Assoc. Comp. Mach.* **32** (1985), 229-246.