

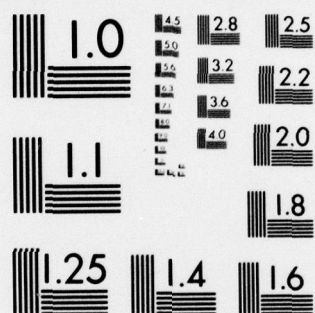
AD-A063 104

MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR COMPUTE--ETC F/G 17/2
ON THE SECURITY OF THE MERKLE-HELLMAN CRYPTOGRAPHIC SCHEME, (U)
NOV 78 A SHAMIR, R E ZIPPEL N00014-76-C-0366
MIT/LCS/TM-119 NL

UNCLASSIFIED

1 OF 1
AD
A063 01

END
DATE
FILMED
3-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

DDC FILE COPY

ADA063104

LABORATORY FOR
COMPUTER SCIENCE



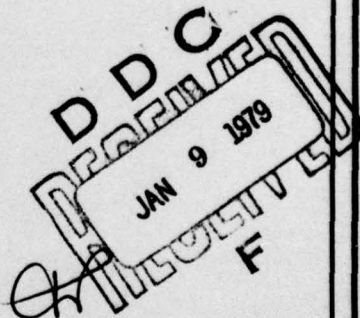
MASSACHUSETTS
INSTITUTE OF
TECHNOLOGY

LEVEL

MIT/LCS/TM-119

ON THE SECURITY OF THE MERKLE-HELLMAN
CRYPTOGRAPHIC SCHEME

Adi Shamir
Richard E. Zippel



December 1978

This research was supported by the Office of Naval
Research under Contract No. N00014-76-C-0366 and
NASA Grant No. NSG 1323

545 TECHNOLOGY SQUARE, CAMBRIDGE, MASSACHUSETTS 02139

79 01 08 065

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER MIT/LCS/TM-119	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) On the Security of the Merkle-Hellman Cryptographic Scheme		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER MIT/LCS/TM-119
7. AUTHOR(s) Adi Shamir and Richard E. Zippel	8. CONTRACT OR GRANT NUMBER(s) N00014-76-C-0366 NSG 1323	
9. PERFORMING ORGANIZATION NAME AND ADDRESS MIT/Laboratory for Computer Science 545 Technology Square Cambridge, MA 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Department of the Navy Information Systems Program Arlington, VA 22217		12. REPORT DATE November 1978
		13. NUMBER OF PAGES 11
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Cryptography public-key cryptosystems Merkle-Hellman knapsacks Graham-Shamir knapsacks		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) In this paper we show that a simplified version of the Merkle-Hellman public-key cryptographic system is breakable. While their full-fledged system seems to be resistant to the cryptanalytic attack we propose, this result suggests some ways in which the security of their system can be further enhanced.		

14
MIT/LCS/TM-119

15
N00014-76-C-0366,
/NSG-1323

6
ON THE SECURITY OF THE MERKLE-HELLMAN
CRYPTOGRAPHIC SCHEME

10
Adi/Shamir
Richard E./Zippel

11
November 1978

12
13 p.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR COMPUTER SCIENCE

CAMBRIDGE

MASSACHUSETTS 02139

ACCESSION for	
NTIS	Write Section ☒
DDC	Bull Section ☐
UNANNOUNCED	☐
JUSTIFICATION	
BY	
DISTRIBUTION/AVAILABILITY CODES	
Dist	SPECIAL
A	

409 648

gm

ON THE SECURITY OF THE MERKLE-HELLMAN
CRYPTOGRAPHIC SCHEME

by

Adi Shamir

and

Richard E. Zippel

Abstract: In this paper we show that a simplified version of the Merkle-Hellman public-key cryptographic system is breakable. While their full-fledged system seems to be resistant to the cryptanalytic attack we propose, this result suggests some ways in which the security of their system can be further enhanced.

Key Words: Cryptography, public-key cryptosystems, Merkle-Hellman knapsacks, Graham-Shamir knapsacks.

This research was supported by the Office of Naval Research under Contract No. N00014-76-C-0366 and NASA Grant No. NSG 1323

79 01 *08 065

1. The Merkle-Hellman Knapsack Systems.

In this section we briefly outline the Merkle-Hellman cryptographic system. A fuller description can be found in [1].

A knapsack system is a vector of n natural numbers $(a_1, \dots, a_n)$. It represents a collection of knapsack problems (or instances) of the following type: given an integer S , find a 0-1 valued vector $(x_1, \dots, x_n)$ such that $S = \sum_{i=1}^n x_i a_i$ (if one exists). Knapsack problems are known to be NP-complete ([2]), and thus they serve as an attractive source for cryptographic functions.

One way of using knapsack systems in public-key cryptography (see [3] for definitions) is to let each network member publish his knapsack system $(a_1, \dots, a_n)$ in a publicly available network directory. Anyone wishing to send an n -bit message $X = (x_1, \dots, x_n)$ to a network member uses the latter's known knapsack system in order to calculate the sum $S = \sum_{i=1}^n x_i a_i$, and to send it over the (insecure) communication channel. An eavesdropper who gets hold of S and who tries to recover X from S is faced with the apparently impossible task of solving the corresponding knapsack problem.

In order to enable the intended receiver of S to solve this knapsack problem, some hidden structure must be embedded in the knapsack system $(a_1, \dots, a_n)$. This structure should be hard to find (i.e., the knapsack system should look like an n -tuple of random numbers to the uninformed observer), but it should enable those who know it to decode encrypted messages quickly by a shortcut method.

The knapsack systems Merkle and Hellman use are based on superincreasing sequences. A vector $(a'_1, \dots, a'_n)$ of natural numbers is a superincreasing

sequence if for each $1 \leq i \leq n$, $a_i' > \sum_{j=1}^{i-1} a_j'$. A simple example of a super-increasing sequence is $(1, 2, 4, 8, \dots, 2^n)$ in which each number equals the sum of its predecessors plus one. Considered as a knapsack system, there is an easy algorithm for solving all the instances of a superincreasing sequence by successive subtractions — see [1] for details.

The numbers a_i' cannot be published in the public directory, since their obvious structure enables any eavesdropper to decode encrypted messages S . To hide this structure, Merkle and Hellman suggest using a modulus m and a multiplier w , such that $m > \sum_{i=1}^n a_i'$ and $\gcd(w, m) = 1$ (this insures the existence of a multiplicative inverse w^{-1} of w modulo m). Instead of publishing a_i' , the network member publishes the numbers a_i , where for each $1 \leq i \leq n$

$$a_i = a_i' \cdot w \pmod{m}.$$

The network member, who knows the unpublished numbers m and w he used, can quickly transform any instance $S = \sum_{i=1}^n x_i a_i$ of the apparently difficult knapsack system $(a_1, \dots, a_n)$ to an instance $S \cdot w^{-1} = \sum_{i=1}^n x_i a_i' \pmod{m}$ of the easily solvable knapsack system $(a_1', \dots, a_n')$, and thus decode S into X . To use this efficient method, a cryptanalyst must determine m and w from the published numbers $(a_1, \dots, a_n)$; the difficulty of this problem is studied in the next section.

In their paper, Merkle and Hellman recommend the following specific parameters for their knapsack systems:

- (i) $n = 100$ (knapsack systems with one hundred elements).
- (ii) Each a_i' is randomly chosen from a uniform distribution over the interval $[(2^{i-1} - 1) \cdot 2^{100} + 1, 2^{i-1} \cdot 2^{100}]$ (it is a $99 + i$ bit natural number).

- (iii) The modulus m is chosen uniformly from the interval $[2^{201} + 1, 2^{202} - 1]$ (thus making all the a_i pseudo-random 202-bit natural numbers).
- (iv) The multiplier w is chosen uniformly from the interval $[2, m-2]$ and then divided by its gcd with m .

2. The Cryptanalytic Attack.

The starting point for our cryptanalytic attack was the following challenge in Merkle and Hellman's paper:

"Attempts to break the system can start with simplified problems (e.g., assuming m is known). If even the most favored of certification attacks is unsuccessful, then there is a margin of safety against cleverer, wealthier, or luckier opponents. Or, if the favored attack is successful, it helps to establish where the security really must reside. For example, if knowledge of m allows solution, then an opponent's uncertainty about m must be large."

In this section we show that the knowledge of m makes any standard-parameter Merkle-Hellman knapsack system highly vulnerable to cryptanalysis.

The key idea is that the first two numbers a'_1 and a'_2 in the unknown superincreasing sequence are much smaller than the modulus m (for the recommended parameters, a'_1 , a'_2 and m are 100, 101 and 202 bits long, respectively). We assume that in the list of published numbers $a_1, \dots, a_n$ the cryptanalyst can identify the two numbers a_1 and a_2 which correspond to a'_1 and a'_2 (if these numbers are published in a shuffled order, the cryptanalyst can repeat the following procedure for each one of the 100·99 possible pairs of published numbers, and still break the system in reasonable time). Since m is known, we can calculate the quotient q :

$$q = \frac{a_1}{a_2} \pmod{m}.$$

But $a_i = a_i' \cdot w \pmod{m}$ and thus

$$q = \frac{a_1' \cdot w}{a_2' \cdot w} = \frac{a_1'}{a_2'} \pmod{m}$$

or

$$a_1' = a_2' \cdot q \pmod{m}.$$

Consider now the set of all the modular multiples of q for multipliers in the range $[1, 2^{101}]$:

$$\{1 \cdot q \pmod{m}, 2 \cdot q \pmod{m}, \dots, 2^{101} \cdot q \pmod{m}\}.$$

Since $a_2' \leq 2^{101}$, $a_2' \cdot q \pmod{m}$ (which is equal to a_1') is in this set. All these 2^{101} multiples are very evenly distributed in the interval $[0, m-1]$, and thus the smallest number among them is likely to be around $m/2^{101} \approx 2^{202}/2^{101} = 2^{101}$. But a_1' is known to be smaller than or equal to 2^{100} , and thus a_1' itself is likely to be the smallest number in this set. Consequently all we need in order to find (a candidate for) a_1' is to find the minimum value of $j \cdot q \pmod{m}$ when j ranges over the interval $[1, 2^{101}]$ and q, m are known. Efficient methods for solving this number-theoretic problem (using the continued fraction approximation of the ratio q/m) can be found in [4] and [5].

Once a candidate value for a_1' is found, w can be calculated as $a_1/a_1' \pmod{m}$ and then the whole sequence a_i' can be generated from m, w and the published numbers a_i . If the candidate value for a_1' is the correct one, the calculated sequence a_i' would turn out to be superincreasing, thus verifying the candidate and giving a quick way of solving instances of the published knapsack system.

It is easy to see that for other choices of the parameters, this

cryptanalytic attack has a good probability of success only as long as $a_1' \cdot a_2'$ is not much larger than m . The network member can of course use Merkle-Hellman knapsack systems in which this condition does not hold.

There are two reasons why such a simple solution might not be adequate:

- (i) If $m > \sum_{i=1}^n a_i'$ and a_i' is superincreasing, then a simple calculation shows that $m \geq 2^n a_1'$ and $m \geq 2^{n-1} \cdot a_2'$, and thus $a_1' \cdot a_2' \leq m^2 / 2^{2n-1}$.

To make $a_1' \cdot a_2'$ much bigger than m in a hundred element knapsack system (which is the minimum secure value), m must have considerably more than 200 bits. This slows down the computations and worsens the ratio between the number of bits in encrypted and original messages.

- (ii) Our cryptanalytic method uses only the two smallest numbers in the superincreasing sequence a_i' . If three or more elements are considered simultaneously, the condition $a_1' \cdot a_2' \leq m$ can be weakened considerably. Although we do not know how to do it at present, it seems dangerous to assume that such an extension is impossible.

3. Safer Variants of the Merkle-Hellman Knapsack Systems.

After defining their basic knapsack systems, Merkle and Hellman note that a safer knapsack system can be obtained by iterating the modular multiplications technique a number of times. At each iteration a new modulus m_j ($m_j > \sum_{i=1}^n a_i$) and a new multiplier w_j ($\gcd(w_j, m_j) = 1$) are chosen, and all the knapsack elements a_i are replaced by $a_i \cdot w_j \pmod{m_j}$. The decoding of encrypted messages is done by successively dividing them by the $w_j \pmod{m_j}$ in the reverse order, thus unwinding the iterations

all the way back to the original superincreasing sequence.

When two or more iterations are used in order to obscure the structure of the superincreasing sequence, our cryptanalytic attack becomes ineffective (even when all the modulus m_j and all but the last w_j are known). The reason is that when we attempt to strip the last w_j from the knapsack elements by dividing pairs of the published numbers modulo the last m_j , we are left with large, random looking numbers (the results of the last but one iteration) to which the minimization technique cannot be applied. In their paper, Merkle and Hellman express the belief that knapsack systems obtained by two iterations are strictly more secure than their simple, single iteration knapsack systems. Our method is an explicit cryptanalytic example which substantiates Merkle and Hellman's intuitive feeling.

Another way of eliminating the potential weakness represented by extremely small knapsack elements has been suggested (independently) by Graham and Shamir. The idea is to use structured numbers, whose low-order parts are a superincreasing sequence and whose high-order parts are strings of random bits:

$$\begin{array}{rcl}
 a'_1 & = & \text{_____} \quad 0 \dots 0 \text{_____} \\
 a'_2 & = & \text{_____} \quad 0 \dots 0 \text{_____} \\
 & & \vdots \quad \vdots \\
 & & \text{(random)} \quad \text{(superincreasing} \\
 & & \quad \quad \text{sequence)} \\
 & & \vdots \\
 a'_n & = & \text{_____} \quad 0 \text{_____}
 \end{array}$$

high-order part
low-order part

Due to the existence of the high-order "noise", none of these numbers is likely to be small, but when some of them are added together, the sum can still be decoded by disregarding its high-order part and analyzing its low-order part in the usual way.

A particularly simple knapsack system is obtained when the low-order part is decomposed further in the following way:

$$\begin{array}{rcl}
 a'_1 & = & \text{-----} \quad 0 \text{ } 1 \quad 0 \text{ } 0 \quad \text{-----} \\
 a'_2 & = & \text{-----} \quad 0 \text{ } 0 \, 1 \, 0 \quad 0 \text{ } 0 \quad \text{-----} \\
 & & \quad \quad \quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 & & \quad \quad \quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 & & \quad \quad \quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 & & \quad \quad \quad \cdot \quad \quad \quad \cdot \quad \quad \quad \cdot \\
 a'_n & = & \text{-----} \quad 1 \, 0 \text{ } 0 \quad 0 \text{ } 0 \quad \text{-----}
 \end{array}$$

$\begin{matrix} \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \end{matrix}$
 $\begin{matrix} \text{(random)} & & \text{(random)} \end{matrix}$

The block of zeros between the low-order random bits and the diagonal matrix is $\log_2 n$ bits wide. Its purpose is to serve as a buffer zone, so that even when all the n numbers a'_i are added together, the sum of the low order bits does not overflow into the region of the diagonal matrix. To obscure this structure, we use $k \geq 1$ iterations of Merkle and Hellman's modular multiplications technique. Encrypted messages are now very easy to decode: once we unwind the iterations back to the a'_i knapsack system, the decoded message can be read off an intermediate interval of bits in the (augmented) encoded message, without any further computations. This variant of Merkle and Hellman's scheme seems to be safer, faster and simpler to implement than the original variant recommended in [1].

Acknowledgements: We would like to thank Len Adleman, Abraham Lempel, Michael Rabin and Ron Rivest for many fruitful discussions.

Bibliography:

- [1] R. Merkle and M. Hellman, "Hiding Information and Receipts in Trap Door Knapsacks", IEEE Trans. Information Theory, September 1978.
- [2] R. Karp, "Reducibility Among Combinatorial Problems", in "Complexity of Computer Computations" (ed. R. Miller and J. Thatcher), Plenum Press, 1972.
- [3] W. Diffie and M. Hellman, "New Directions in Cryptography", IEEE Trans. Information Theory, November 1976.
- [4] J. Cassels, "An Introduction to Diophantine Approximation", Cambridge University Press, 1965.
- [5] W. LeVeque, "Fundamentals of Number Theory", Addison-Wesley, 1977.

OFFICIAL DISTRIBUTION LIST

Defense Documentation Center
Cameron Station
Alexandria, VA 22314
12 copies

Office of Naval Research
Information Systems Program
Code 437
Arlington, VA 22217
2 copies

Office of Naval Research
Branch Office/Boston
495 Summer Street
Boston, MA 02210
1 copy

Office of Naval Research
Branch Office/Chicago
536 South Clark Street
Chicago, IL 60605
1 copy

Office of Naval Research
Branch Office/Pasadena
1030 East Green Street
Pasadena, CA 91106
1 copy

New York Area Office
715 Broadway - 5th floor
New York, N. Y. 10003
1 copy

Naval Research Laboratory
Technical Information Division
Code 2627
Washington, D. C. 20375
6 copies

Assistant Chief for Technology
Office of Naval Research
Code 200
Arlington, VA 22217
1 copy

Office of Naval Research
Code 455
Arlington, VA 22217
1 copy

Dr. A. L. Slafkosky
Scientific Advisor
Commandant of the Marine Corps
(Code RD-1)
Washington, D. C. 20380
1 copy

Office of Naval Research
Code 458
Arlington, VA 22217
1 copy

Naval Electronics Lab Center
Advanced Software Technology
Division - Code 5200
San Diego, CA 92152
1 copy

Mr. E. H. Gleissner
Naval Ship Research & Development Center
Computation & Math Department
Bethesda, MD 20084
1 copy

Captain Grace M. Hopper
NAICOM/MIS Planning Branch
(OP-916D)
Office of Chief of Naval Operations
Washington, D. C. 20350
1 copy

Mr. Kin B. Thompson
Technical Director
Information Systems Division
(OP-91T)
Office of Chief of Naval Operations
Washington, D. C. 20350
1 copy

Captain Richard L. Martin, USN
Commanding Officer
USS Francis Marion (LPA-249)
FPO New York, N. Y. 09501
1 copy

RECORDING PAGE NOT FILMED
BLANK

