"CRACKING" A RANDOM NUMBER GENERATOR

James Reeds

[Editor's Note: This paper was original printed in the January 1977, Volume I, Number 1, issue of CRYPTOLOGIA (pp. 20-26) - the premier issue. Over the years this one article has drawn more interest than any other article and requests for reprints of the paper come in year after year. The editors thought it appropriate to offer this paper to our readers. The original paper was set using WordStar word process in a NEC Spinwriter thimble printer. We have set this version using LaTeX on a laser printer, the configuration we currently use to do the journal itself. Times have changed, but the lure of this piece remains. James Reeds has written much over the years about crypto maters, indeed, his papers have appeared in CRYPTOLOGIA since this very first one appeared. We hope you enjoy the work.]

The purpose of this note is to illustrate how the ordinary standards of randomness have little to do with the type of randomness required for cryptographic purposes. That is, there are really two standards of randomness.

I. Consider the usual standards for random number generators. Here the general idea is that standard techniques of statistical analysis are not able to discriminate between the sequence of numbers generated and a sequence of independent uniform deviates from the unit interval. Thus, χ^2 tests, autocorrelation functions, and correlation coefficients are used to judge the random number generator in question. These are discussed at length in [1]. The point of this standard is that acceptable random number generators should be suitable for "Monte Carlo" applications.

II. On the other hand we have the standards of cryptography. For cryptographic purposes the matter of predictability is exceedingly important. If, after examining, say, a sequence of four random numbers, one is able to predict the fifth (and all subsequent) number, then that generator is useless for cryptographic purposes. In predicting the next number we are allowed to examine the low-order bits (or digits) as well as the high-order bits. As a result, the "rule" which predicts the next number may be "discontinuous", and thus not be discovered by the standard statistical methods used to evaluate "randomness I" properties of

a random number generator.

As an illustration of what I mean, let us examine a "typical" cryptographic example. Let us say that a secret message has been prepared by converting the letters into digits, following the rule $A = 01$, $B = 02$, etc., to $Z = 26$. Then the successive digits are added, modulo 10 to the successive digits of the output of a "linear congruential" random number generator. The correspondents have previously agreed upon a "modulus" $M = 8397$, a "multiplier" $a = 4381$, and a constant term $b = 7364$. That is, if x_n is the n^{th} random number, the next is given by the rule:

$$x_{n+1} \equiv 4381x_n + 7364 \bmod 8397.$$

(I chose these three numbers entirely at random, insisting only that they have four digits. The reader will see how the analysis given below is general and will apply to other choices of M, a, and b.)

Let us assume further that the correspondents have agreed (ahead of time) to encipher this message by starting up the random number generator with the "initial key" of $x_0 = 2134$. With the generator given above, we get:

$x_0 = 2134$	$x_4 = 8295$	$x_8 = 7907$	$x_{12} = 7648$	$x_{16} = 6636$
$x_1 = 2160$	$x_5 = 5543$	$x_9 = 0766$	$x_{13} = 0825$	$x_{17} = 0869$
$x_2 = 6905$	$x_6 = 7123$	$x_{10} = 3231$	$x_{14} = 2582$	$x_{18} = 2215$
$x_3 = 3778$	$x_7 = 1578$	$x_{11} = 1865$	$x_{15} = 8347$	$x_{19} = 4347$

These successive digits (starting with x_1: 2160, 6905 etc.) are added (modulo ten) to the message digits to get the cryptogram:

plaintext	S	E	C	R	E	T	T	R	E	A	T	Y	
plaintext digits	19	05	03	18	05	20	20	18	05	01	20	25	
key digits	21	60	69	05	37	78	82	95	55	43	71	23	
ciphertext	30	65	62	13	32	98	02	03	50	44	91	48	

S	I	G	N	E	D	B	Y	P	A	K	I	S	T	A	N	A	N	D
19	09	07	14	05	04	02	25	16	01	11	09	19	20	01	14	01	14	04
15	78	79	07	07	66	32	31	18	65	76	48	08	25	25	82	83	47	66
24	77	76	11	02	60	34	56	24	66	87	47	17	45	26	96	84	51	60

I	S	R	A	E	L
09	19	18	01	05	12
36	08	69	22	15	43
35	17	77	23	10	55

Now, it is not suggested that the preceding cipher system is a good system, or an especially practical one, or a widely used one. I show below why it is not a good system, and I doubt if it is widely used. The point is that very similar systems might well be in use in computers. The "linear congruential" random

number generator is by far and away the most popular generator in the computer world, and similar cipher systems (based on bits, not digits) might well be used with computers. In such a computer system the correspondence between letters and bits is provided by one of the standard codes: Baudot, ASCII, or EBCDIC.

Let us now assume that this message is intercepted by a cryptanalyst. He does not know the starting random number x_0, nor the modulus, nor the multiplier, nor constant terms, M, a, and b. what he does know (from the study of similar messages) is that the numbers are 4 digits long. Further, he suspects that the word "Pakistan" occurs in the messages. He uses the *probable word method* to recover the key digits, and mathematical analysis to reconstruct the generator. He tries to fit the word "Pakistan" in at the beginning of the message, gets "false" key digits, and hence the wrong generator. This wrong generator does not yield any intelligible text, so the cryptanalyst tries another place to fit "Pakistan" in. Place by place the analysis is followed, and time after time no intelligible text is produced. Finally, however, "Pakistan" is fitted into the correct place, and at last the "true" key digits are produced. The analysis is as follows:

The cipher text has been lined up with the digits for "Pakistan", and the probable word has been subtracted out, modulo 10:

ciphertext:	24	66	87	47	17	45	26	96	84	51	60	etc.
probable word:	16	01	11	09	19	20	01	14				
key:	18	65	76	48	08	25	25	82				

By blocking off digits by fours from the beginning of the message we get four consecutive 4-digit numbers: 1865, 7648, 0825, 2582. The cryptanalyst tries to recover the entire random number generator from these data.

It is clear that the modulus M is at least as large as 7,649 (and, by the rules of this cipher system, no greater than 10,000). Referring back to the equation defining the "linear congruential" system, we get:

$$\begin{aligned} 7648 &\equiv 1865a + b \bmod M && \text{(I)} \\ 825 &\equiv 7648a + b \bmod M && \text{(II)} \\ 2582 &\equiv 825a + b \bmod M && \text{(III)} \end{aligned}$$

Take equation I and subtract it from II and III, to get:

$$\begin{aligned} -6823 &\equiv 5783a \bmod M && \text{(IV)} \\ -5066 &\equiv -1040a \bmod M. && \text{(V)} \end{aligned}$$

Thus, b is eliminated from these equations. Now we try to eliminate a. We can find no common factor (other than 1 and -1) of the two numbers 5783 and -1040, so in order to eliminate a from IV and V we have to multiply IV by 1040 and V by 5783 and add the two together. Thus, we get:

$$-36,392,598 \equiv 0 \bmod M. \qquad \text{(VI)}$$

This lets us say that M divides 36,392,598, and, if we list all the divisors of this large number, M will be found among them. So we must factorize the number $N = 36,392,598$. (This can be done automatically on a computer, but it is fun to do by hand.) First off, it ends in an even digit, so we can divide out 2: $N = 36,392,598 = 218,196,299$. The sum of the digits is 45, so 3 divides N: $N = 2 \cdot 3 \cdot 6,065,433$. Again, 3 divides 6,065,433: $N = 2 \cdot 3 \cdot 3 \cdot 2,021,811$, and again: $N = 2 \cdot 3 \cdot 3 \cdot 3 \cdot 673,937$.

Now we try trial divisors of primes higher than 2 and 3: 5, 7, 11 etc. We find 5 and 7 don't divide 673,937, but 11 does: $N = 2 \cdot 3^3 \cdot 11 \cdot 61,267$. We try 11 again, and all of the primes lower than the square root of 61,267, i.e., less than 247. We find that the lowest prime divisor of 61,267 is 197, which goes in 311 times. This last number is (by reference to a table of primes) seen to be a prime, and so we have

$$N = 2 \cdot 3^3 \cdot 11 \cdot 197 \cdot 311$$

as the complete factorization of N. There are 64 possible divisors, but many are too big (i.e., larger than 10,000), and others are smaller than 7,649.

Look at Figure 1, a table of size 8×8, showing all the possible divisors of N:

		Values of B							
		1	2	3	$2 \cdot 3 = 6$	$3 \cdot 3 = 9$	$2 \cdot 3 \cdot 3 = 18$	$3 \cdot 3 \cdot 3 = 27$	$2 \cdot 3 \cdot 3 \cdot 3 = 54$
Values of A	1								
	11								
	197								
	311								
	$11 \cdot 197 = 2,167$								
	$11 \cdot 311 = 3,421$								
	$197 \cdot 311 = 61,267$								
	$11 \cdot 197 \cdot 311 = 673,937$								

Figure 1.

A possible divisor is formed by picking a cell, and multiplying the numbers A and B standing at the ends of the row and column that meet at the cell in question. This product may be entered in the cell.

Before we begin, we see that we may rule out many entries in the last two rows in the table, because these products will be greater that 10,000.

	1	2	3	6	9	18	27	54
1	x	x	x	x	x	x	x	x
11	x	x	x	x	x	x	x	x
197	x	x	x	x	x	x	x	o
311	x	x	x	x	x	x		o
2,167	x	x	x	o	o	o	o	o
3,421	x	x	o	o	o	o	o	o

We similarly rule out the o'ed regions ($A \cdot B$ is too big) and the x'ed regions ($A \cdot B$ is too small). In fact, there is only one divisor of N left (in the range 7649 through 10,000), and it is $27 \cdot 311 = 8,397$. This is thus the only candidate for M.

Referring back to the original generator, we see that this is indeed correct, we can now try to solve equation IV for a:

$$-6823 \equiv 5783a \bmod 8397.$$

Without going through the calculation, we can check to see if this is in fact solvable. If 5,783 has no common factor with 8,397, there is a unique solution. Well, 3 doesn't divide 5,783, and neither does the prime 311, so they are in fact relatively prime, and thus a unique solution exists. (It can be found by application of Euclid's algorithm for finding the G.C.D. of two numbers.)

Once a is found, the cryptanalyst can solve Equation I for b:

$$b \equiv 7648 - 1865a \bmod 8397.$$

At this point, the cryptanalyst has recovered the generator, and he can crank out the next several numbers to decipher the words following "Pakistan". He finds, of course, "and Israel". This makes sense (linguistically, if not politically!) and the cryptanalyst knows he has the right key. Since $a = 4381$ is relatively prime to $M = 8397$, the random number generator may be "cranked" backwards to yield the previous parts of the keying sequence, and the whole message may be read.

This may all, of course, be done automatically on a computer. The computer will try a probable word in each of the possible places it could fit in the message, go through the calculations outlined above very rapidly, and then print out a portion of the resulting supposed plain texts. The cryptanalyst could quickly scan the list of trial decipherments and pick out the correct one. Of course, if such a method of encipherment were ever to be used, it would be based on much larger numbers: of 10 instead of 4 digits, and its decipherment would be a bit more difficult, especially at the factorization step. But this is nothing a good computer could not handle.

Thus, we've seen how "linear congruential" random number generators are unsuitable for cryptographic applications. The method presented above is applicable against any linear congruential generator, and is not affected in the least by whether or not the generator is judged highly random or not by "standard I" criteria mentioned at the beginning of this note. Moreover, the general idea of the analysis presented in this note may be carried over to other random number generators, including the "squaring the middle half" and "shift register sequence" generators, for instance. That is to say, cryptography has its own standards of randomness, which do not necessarily coincide with the more usual standards.

REFERENCES

1. Knuth, Donald E. 1969. *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms*. Reading MA: Addison-Wesley Publishing Company. pp. 1-99.

BIOGRAPHICAL SKETCH

James Reeds received his AB (The University of Michigan, 1969) and MA (Brandeis, 1972) in mathematics and his PhD (Harvard, 1976) in statistics. He will be teaching statistics at the University of California, Berkeley. He has always been interested in cryptanalysis, and after reading *The Codebreakers* in college he began using mathematics and computers in cryptanalysis. He is most interested in statistical methods for breaking machine ciphers.