The distinguishing attack on ZK-Crypt cipher *

Alexey Lubkin† and Boris Ryabko ‡

† Siberian State University of Telecommunications and Computer Science, Russia

‡ Institute of Computational Technologies of Siberian Branch of Russian Academy of Science

Abstract

Stream cipher ZK-Crypt is a submission to the ECRYPT stream cipher project. The "book stack" test for randomness from [2] was applied to this cipher. It is experimentally shown that the keystream generated from ZK-Crypt can be distinguished from random with about 2^{25} output bits.

1 Introduction.

ZK-Crypt [1] is a stream cipher proposed as a candidate to ECRYPT Stream Cipher Project. It is required that a keystream of any stream cipher should be truly random, i.e., by definition, a generated sequence could be interpreted as the result of the flips of a "fair" coin with sides that are labelled "0" and "1" (for short, it is called a random sequence).

In this paper, we describe some experiments intended to detect non-randomness of the ZK-Crypt keystream. In other words, we describe a distinguishing attack on ZK-Crypt cipher, whose goal is to distinguish the keystream of the cipher from the random sequence. For this purpose the "book stack" test from [2] was applied to keystream sequences generated by ZK-Crypt cipher.

It turns out that the keystream sequences are far from random when their length is about 2^{25} bits.

A description of the experiments is given in section 2. The Appendix contains a brief description of the book stack test and its parameters.

2 Experiments.

The behavior of the ZK-crypt keystream was investigated for 100 randomly chosen keys by the book stack test from [2]. When the test was applied, the keystream sequence was divided into 32-bit words (blocks) and any block was considered as a letter from an alphabet whose size is 2^{32}. According to the test, any alphabet letter

*Research was supported by Russian Foundation for Basic Research (grant no. 03-01-00495).

(i.e. a 32-bit word) was assigned an index and the set of all indexes was divided into two subsets $A_1 = \{1, ..., 2^{16}\}$ and $A_2 = \{2^{16} + 1, ..., 2^{32}\}$ (a brief description of the test is given in Appendix and can be found in [2]).

We generated files of different lengths for each key (see the tables below) and applied the book stack test to each file with level of significance 0.001. So, if a test is applied to a random bit sequence, on average 1 file from 1000 should be recognized as non-random. All results are given in the table, where integers in the cells are the numbers of files recognized as non-random (out of 100).

Table 1: Number of files generated by ZK-Crypt and recognized as non-random (from 100).

Length (bits)	16777216= 2^{24}	67108864= 2^{26}	268435456= 2^{28}	1073741824= 2^{30}
Non-random	25	51	97	100

Having taken into account that, on average, only 0.1 files from 100 should be recognized as non-random if sequences are random, we see that the keystream sequences are far from random when their length is about 2^{25}.

3 Appendix.

Here we give a short description of the book stack test from [2].

Let there be given an alphabet $A = \{a_1, ..., a_S\}$, a source, which generates letters from A, and two following hypotheses: the source is i.i.d. and $p(a_1) = ... = p(a_S) = 1/S$ (H_0) and $H_1 = \neg H_0$. (In the above described experiments $S = 2^{32}$ and A is the set of all 32-bit words.) One should test the hypotheses basing on a sample $x_1 x_2 ... x_n$, $n \geq 1$, generated by the source. When the "book stack" test is applied, all letters from A are ordered from 1 to S and this order is changed after observing each letter x_t according to the formula

$$\nu^{t+1}(a) = \begin{cases} 1, & \text{if } x_t = a\,; \\ \nu^t(a) + 1, & \text{if } \nu^t(a) < \nu^t(x_t); \\ \nu^t(a), & \text{if } \nu^t(a) > \nu^t(x_t)\ , \end{cases} \tag{1}$$

where ν^t is the order after observing $x_1 x_2 ... x_t$, $t = 1\,, , ... , n\,$, ν^1 is defined arbitrarily. (For ex., we can define $\nu^1 = \{a_1, ..., a_S\}$.) Let us explain (1) informally. Suppose that the letters of A make a stack, like a stack of books and $\nu^1(a)$ is a position of a in the stack. Let the first letter x_1 of the word $x_1 x_2 ... x_n$ be a. If it takes i_1-th position in the stack ($\nu^1(a) = i_1$), then take a out of the stack and put it on the top. (It means that the order is changed according to (1).) Repeat the procedure with the second letter x_2 and the stack obtained, etc.

It can help to understand the main idea of the suggested method if we take into account that, if H_1 is true, then frequent letters from A (as frequently used books)

will have relatively small numbers (will spend more time next to the top of the stack). On the other hand, if H_0 is true, the probability to find each letter x_i at each position j is equal to $1/S$.

The set of all indexes $\{1, \ldots, S\}$ is divided into $r, r \geq 2$, subsets $A_1 = \{1, 2, \ldots, k_1\}$, $A_2 = \{k_1 + 1, \ldots, k_2\}, \ldots, A_r = \{k_{r-1} + 1, \ldots, k_r\}$. Then, using $x_1 x_2 \ldots x_n$, we calculate how many $\nu^t(x_t)$, $t = 1, ..., n$, belong to a subset $A_k, k = 1, ..., r$. We define this number as n_k (or, more formally, $n_k = |\{t : \nu^t(x_t) \in A_k, t = 1, \ldots, n\}|, k = 1, ..., r.$) Obviously, if H_0 is true, the probability of the event $\nu^t(x_t) \in A_k$ is equal to $|A_j|/S$. Then, using the chi- square test we test the hypothesis $\hat{H}_0 = P\{\nu^t(x_t) \in A_k\} = |A_k|/S$ basing on the empirical frequencies $n_1, \ldots, n_r$, against $\hat{H}_1 = \neg\hat{H}_0$. Let us recall that the value

$$x^2 = \sum_{i=1}^{r} \frac{(n_i - n(|A_i|/S))^2}{n(|A_i|/S)} \tag{2}$$

is calculated, when chi- square test is applied. It is known that x^2 asymptotically follows the χ-square distribution with $(r-1)$ degrees of freedom (χ^2_{r-1}) if $\hat{H}_0$ is true. If the level of significance (or a Type I error) of the χ^2 test is $\alpha, \alpha \in (0, 1)$, the hypothesis $\hat{H}_0$ is accepted when x^2 from (2) is less than the $(1-\alpha)$ *-value* of the χ^2_{r-1} distribution.

The authors of [2] do not describe the exact rule how to construct the subsets $\{A_1, A_2, \ldots, A_r\}$, but they recommend to perform some experiments for finding the parameters, which make the sample size minimal (or, at least, acceptable). The point is that there are many cryptographic and other applications where it is possible to implement some experiments for optimizing the parameter values and, then, to test hypothesis basing on independent data. Thus, in case of testing a stream cipher it is possible to seek suitable parameters using some keys and then to test the keystream using different keys.

Consider a simple example. Let $A = \{a_1, \ldots, a_6\}$, $r = 2, A_1 = \{a_1, a_2, a_3\}, A_2 = \{a_4, a_5, a_6\}$, $x_1 \ldots x_8 = a_3 a_6 a_3 a_3 a_6 a_1 a_6 a_1$. If $\nu_1 = 1, 2, 3, 4, 5, 6$, then $\nu_2 = 3, 1, 2, 4, 5, 6$, $\nu_3 = 6, 3, 1, 2, 4, 5$, etc., and $n_1 = 7, n_2 = 1$. We can see that the letters a_3 and a_6 are quite frequent and the "book stack" indicates this nonuniformity quite well. (Indeed, the average values of n_1 and n_2 equal 4, whereas the real values are 7 and 1, correspondingly.)

Let us consider the complexity of the algorithm. The "naive" method of transformation according to (1) can take the number of operations proportional to S, but there exist algorithms, which can perform all operations in (1) using $O(\log S)$ operations.

4 Acknowledgment

The authors wish to thank Viktor Monarev who independently repeated experiments described in the part 2 and obtained close results.

References

[1] Carmi Gressel, Ran Granot and Gabi Vago. *ZK-Crypt.* ECRYPT Stream Cipher Project Report 2005. Available at http://www.ecrypt.eu.org/stream/zkcrypt.html

[2] Ryabko B., Monarev V., *Using Information Theory Approach to Randomness Testing.* Journal of Statistical Planning and Inference, 133(1) 95-110, 2005.