International Journal of Advanced Technology in Engineering and Science
Vol. No.5, Issue No. 07, July 2017
www.ijates.com
ijates
ISSN 2348 - 7550

Predicting a PRNG Value in Stream Bits: A practitioner Approach

M.Ravi

Kalasalingam University, Krishnankoil, Tamilnadu (India)

ABSTRACT

Hacking and phishing are dominant the cryptographic worlds based on the tracing and brutal attack. In every cryptographic algorithms has some holes to break the contents. Tracing Pseudo Random Numbers Generation (PRNG) in every algorithm lead the way (hole-leakage) to predict the stream bits. Streaming contents are having weak contents to predict the bits even-though they have long bit-processes. Random Bit Generator (RBG) or Random Number Generator (RNG) is the prime ingredients to injecting the bits in various streaming algorithms like RC4.

Key: streaming algorithms, RC4, PRNG, RBG

I. INTRODUCTION

In the year of 1940 C.E.Shannon proposed a wonderful theorem about the cryptography entitled as 'information theory' which tells what is mean by cryptography? It leads to a way to find one time pad ciphers. C.E.Shannon generated only numbers not binary digits. There find the next bit is very hard, compared with binary digits (bits).

After that, a group of three members Rivest, Shamir, et.al, making standard cryptographic algorithm to competitive with another best algorithm known as DES. RC4 is one hereditary of RSA which was invented by Rivest as streaming cipher. Like that every streaming algorithm has its own story. Identify weakness of an algorithm based on their origin is one of the way.

National Institute of Standard Technology (NIST) was regulated the terms and conditions for cryptographic algorithms to strengthening security. They derive some various testing protocols to reach effective and efficient cryptographic algorithms [1], based on the testing we predict some bits. This will be apply as second category of my prediction test.

The third segment is to apply some attacking algorithm such as brute-force attack, B Sadaghiyan-J mohajeri [] test to getting the real bit streams.

II. RC4 ALGORITHM

The RC4 algorithm [2] is still we are using for real time streaming videos even though it has some weakness [2]. Here RC4 has been taken as stream algorithm for attacking or analysis of my research to find the PRNG values. RC4 has two segments for its security called as KSA and PRGA [2], the out of PRGA values are converted into binary digits (bits) and those values are used for prediction testing.

III. WEAKNESS OF RC4

Weakness of RC4 has already explained by many authors [5] [6] [7] [8]. In KSA segments RC4 revealed its contents while generated. The first author [5] says that we will find that biases (it may 1 or 0) are due to the non-uniformly distributed S. That means the initialization RC4 is weak.

He detailed explain how to find the 1 in 1/n numbers based on Cipher only attack. Literally speaking to find we have n numbers as key stream we are trying to predict i, i+1, i+2 values. Where the algorithm says whether is it good or bad.

IV. ATTACKING AND TESTING

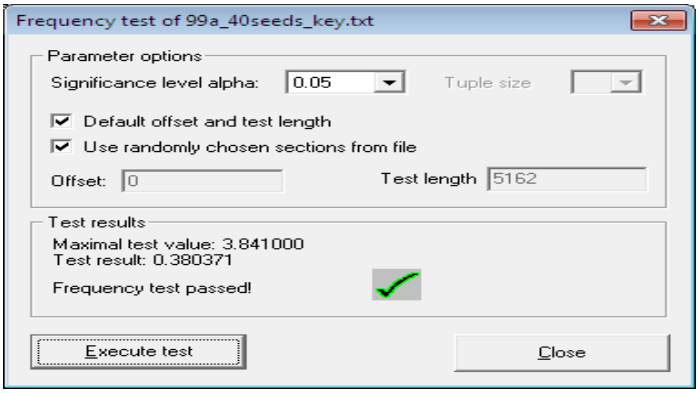
In KSA $a_{0,1,\ldots,i\ldots,j,\ldots,n}$ is the binary sequences while predicting i value by cipher based attack. There is another old method to find 'i' and 'i+1' value. Discrete Logarithm Problem to find out the random numbers. Before that we move with the legacy Random Bit Generator (RBG) now it is called as PRNG [3]. The bits of n numbers easy to generate and polynomial. But generate as Random is comparative unique in the early stage but modern world we just put Random function in any programming languages.

A and B want to play head and tail in 4 different ways. In all of them A "fairly" flips a "fair" coin. In the first way. A asks B to bet and then flips the coin. In such a case we expect B to win with a 50% frequency. In the second way. A flips the coin and. while it 'is spinning in the air. She asks B to bet. We are still expecting B to win with a 50% frequency", however, in the second case the outcome of the toss is determined when B bets: in principle, he could solve ~he equation of the motion and win! The third way is similar to the second one: B is allowed to bet when the coin is spinning in the air. But he is also given a pocket calculator. Nobody will doubt that in this case B is going to win with 50% frequency. As while he is still initializing any computation the coin will have come up head or tail. The fourth way is similar to the third. Except that now B is given a very powerful computer. Able to take pictures of the spinning coin, and quickly compute its speed, momentum etc. In such a case we will not say that B will always win, but we may suspect he may win 51% of the time!

This is theme to apply on RC4 cipher's stream by NIST suggested frequency test and poker test to grab the binaries

{0 1 0 0 1 1 0 1} {1 1 1 0 0 0 1 1} {1 0 0 0 0 1 0 0} {0 0 0 1 0 1 1 0} {0 1 1 1 0 1 0 0} {0 0 0 0 1 1 1 0}

Based on the above byte sequences just apply frequency test such as 00,01,10,11,000, etc., is repeated and how much time is repeated tested by NIST suggested tool called cryptool as below.

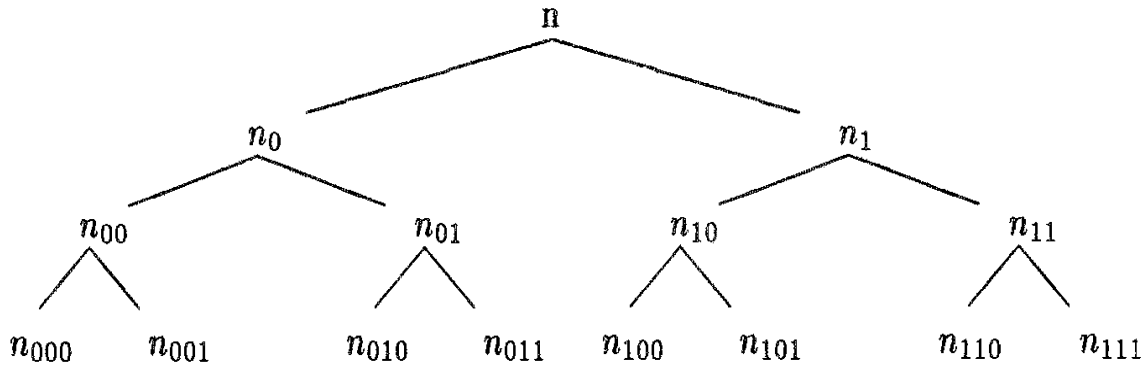


1. Frequency test for RC4 40 bit seeds

Behalf of RC4 256 bit rounds below 123 keys are advisable and we can use 2048 bit rounds but it will lead to some malpractice such as hacking and phishing the stream.

V. NEXT BIT PREDICTION TEST

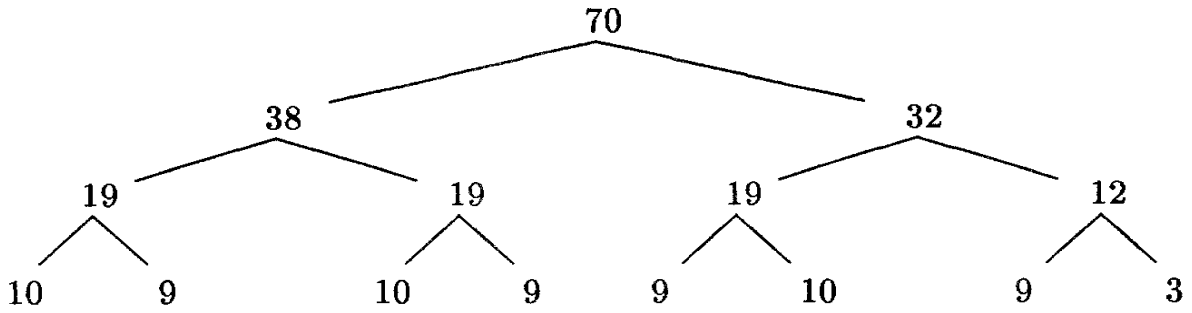
The NBP test initiated by Sadeghiyan, Mohajeri in 1996. After authors [4] explain the modified concept of NBP. It has also pros and cons.



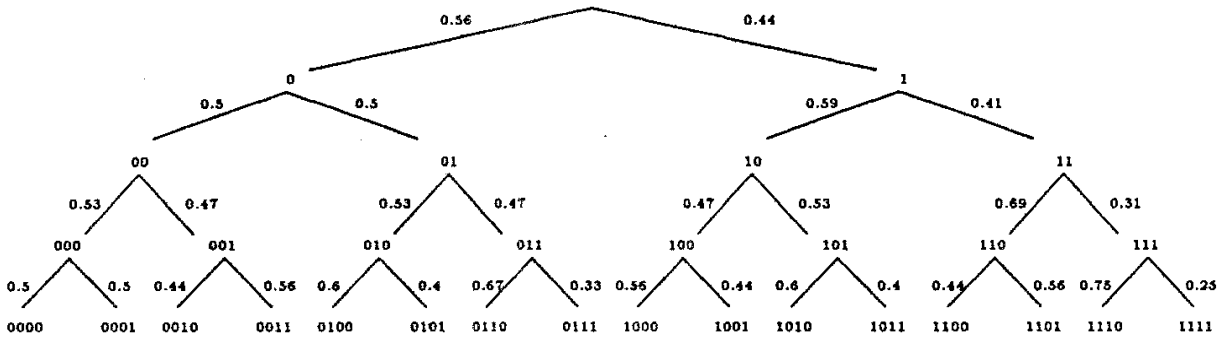
In more detail

Consider the following string and the related tree, where n = 70:

1100110101100101000110100010000011011011010101000011100111000001001011



In technically by the PNB test



This architecture was designed based on the following algorithm

Definition 1: An ensemble S is a sequence {S~}, where S~ is a probability distribution on {0, 1}".

Definition 2: An ensemble S is called *uniform,* if S, ~ is the uniform probability distribution for every n, that is: semble S is called *uniform,* if S, ~ is the uniform probability distribution for every n, that is:

$$Prob\{s_1^n = \alpha\} = \frac{1}{2^n}$$

for every as E {0, 1} n. We denote the uniform ensemble by U.

VI. ALGORITHM

1. Calculate the threshold of decision, a (alpha), according to the formula:

$$\alpha = \frac{1 + \sqrt{\frac{\chi^2}{n}}}{2}$$

2. Calculate l = *round (log2 (n)).*
3. Append l - 1 bits of the beginning of the string to its tail, and divide the string into overlapping g-bit sections.
4. Count the number of occurrence of each pattern of length l.
5. Form part of the tree at layers l- 1 and g, and write down the corresponding probabilities on each edge.
6. For each node at layer l- 1, if the next bit (either 1 or 0) appears with a probability more than a (alpha) then the next bit is predicted accordingly, otherwise the next bit cannot be decided.
7. For each node at layer l- 1, calculate the length of the string which can be predicted after it.

VII. CONCLUSION AND FUTURE WORK

The bit prediction was done based on some NIST suggested algorithms. In future i will completely testing stream bit whether it has full security based on all the algorithms of suggested by NIST. Up to the 2017 the NIST suggested 16 kinds of test and its tool for finding best algorithms for stream cryptography. The authors [] was recommended that we can only predict up to 51% bits of a series. Simultaneously I will develop an algorithm which predicts more 51% bits in a bit stream.

REFERENCES

[1] Rukhin, A. Juan Soto, et.al. "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications", NIST special Publication 800-22, 2001.

[2] Scott Fluhrer, Itsik Mantin, and Adi Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4", 2001.

[3] M. Blum and S. Micaali, "How to Generate Cryptographically Strong Sequences of Psuedo Random Bits" IEEE Xplore, [FOCS] 1982.

[4] A. Lavasani and T. Eghilidos, "Practical Next Bit Test for Evaluating Pseudorandom Sequences" Scientia Iranica, 2009.

[5] Jing LV, et.al. "Some new Weakness in the RC4 Stream Cipher" Springer international publishing. 2014.

[6] Mantin, I, Shamir A, "A practical attack on broadcast RC4.In" Matsui M, (ed.) FSE 2001 LNCS, Vol 2355 Springer 2002.

[7] Isobe T, Ohigashi T "Full plaintext recovery attack on broadcast RC4" FSE 2013, 2013.

[8] Maitra S, Paul G, S Gupta "Attack on broadcast RC4 revisited" FSE 2011, 2011.