ISMSIT2017

1st International Symposium on Multidisciplinary Studies and Innovative Technologies Proceedings Book

(November 2-4, 2017, Tokat, Turkey)

Symposium homepage: www.ismsitconf.org

Web Based AIS31 Test Software For Random Number Generators

Erdinç Avaroğlu[1*], Recai Sinekli[2], Veli Yıldız[3] and Hakan Gülcan[4]

[1]*Engineering Faculty, Mersin University, Turkey*

[2]*Engineering Faculty, Mersin University, Turkey*

[3]*Engineering Faculty, Mersin University, Turkey*

[4]*Engineering Faculty, Mersin University, Turkey*

[*]*(eavaroglu@mersin.edu.tr)*

Abstract – Several test suites have been developed to test the statistical properties of random numbers generated from pseudo-true random number generators. Some of these are Crypt-X, Diehard, FIPS 140, NIST 800-22 AIS31 test suites. In this study, web based application of AIS31 test suite was realized and published at http://ais31.mersin.edu.tr. For this purpose, a random number was generated using the Random () function in the Linux operating system, and it has been shown that the random number sequences obtained from this generator pass the AIS31 test successfully.

Keywords – Random number, Random number generation, NIST statistical test, AIS 31 test, Web based AIS31

I. Introduction

Random numbers can be expressed as numbers that have no specific relationship between them, are equal to each other, and are defined specifically for a given range. Random numbers are widely used today, from cryptography to simulation, from numerical analysis to sampling, from fun to computer programming. Random numbers should have unpredictable, non-reproducible, good statistical properties and uniform distribution in terms of their general characteristics [1].

Various number generators have been developed to obtain random numbers. These are:

- Pseudo random number generators (PRNG)
- True random number generators (TRNG)
- Hybrid random number generators (HRNG)

The long number sequences produced by the pseudo random number generators are generated by subjecting them to an algorithm determined by any initial (seed) value. True random number generators take real physical processes as noise sources and use them to generate random numbers [2].

A variety of test packages have been developed to test the randomness of the random number sequences generated by these number generators, such as FIPS140, Knuth, Ent, NIST, Diehard, Crypt-X and AIS31. Of these test packages, FIPS140 and NIST test packages are the most preferred. However, when the academic publications published in recent years are examined, it has been observed that the popularity of the AIS31 test package has increased.

In addition, the AIS31 test suite software is difficult to use because, in our review, we do not have enough knowledge of computer usage in the software provided in the AIS31 Test Suite [3], and additional software packages are required to run the software in the computer environment. Due to these reasons, many users who need to use the randomness tests cannot carry out the tests. In order to avoid such problems, the AIS31 software has been developed on the web, where researchers who need to apply tests can easily access, use and test the data they produce. Random number data was obtained with the / dev / random function in Ubuntu 16.04 LTS operating system. The statistical tests of the random number sequence generated by the function have been realized with web based developed software according to AIS31 test package standard and the web based software has been shown to work successfully.

In Chapter 2, we discuss the methods used for generating random numbers and the Linux functions used. In Chapter 3, specific explanations have been made to the AIS31 test package. The results of the AIS31 test package of the obtained number sequence are given in Chapter 4. Finally, Chapter 5 gives the conclusion.

II. Random Number Generation Methods

Random number generation is performed with random number generators. Unpredictability, lack of correlation, non-repetition and statistically independent number arrays are among the main features of random number generators. RNGs can be thought of as systems or devices. These systems or devices have a crucial position in modeling and sampling complex phenomena, in numerical analysis and in decision making. For these numbers, the storage area should not be consumed too much and should not be used to reach the result. These figures may not be sufficiently random for another application, while the numbers provided in the light may be sufficiently "random" for one application [2].

Random number generators are divided into 3. These; pseudo-random, true random, and hybrid random number generators.

First, the pseudo-random number generators use the seed value as the random input. Because of the deterministic nature of these generators, the output value produced cannot be

greater than the initial seed value. In addition, the series begin to repeat themselves after a certain period of time. This shows the periodicity of the system. If the periodicity feature is not desired, it is necessary that the memory allocated to the system needs to be large. But this leads to the slowness of the system. The safety of the system in PRNGs depends directly on the inability to predict the seed value initially used and the complexity of the functions to be used in the system. It is also important how the parameters used in these functions are selected. The main algorithms used in these generators are; linear feedback regulator, the medium square method and the linear equivalent method [2], [4], [5].

In true random number generators, non-deterministic natural physical events are defined as seed values and then used in random number generation. These events usually involve conversion to analogue-noise digital signals. The characteristics and the degree of randomness of the random number sequences that are produced by TRNGs pass through the randomness of the physical processes. It is seen as disadvantages of this system being costly, navy dependent and slow. However, it has taken place in many applications because of the unpredictability, non-reproducibility and good statistical properties of random numbers generated by TRNGs [2], [4], [5].

The use of the random number sequence obtained from the TRNG as an entropy source in PRNG and the random number generators that these two systems work together are called hybrid random number generators. The hybrid random number generator generates a random number by which both systems work together by using the seed value in the PRNG of the random number obtained from the TRNG.

In order to generate random numbers, functions such as Perl script, Unix system information collection, Linux / dev / random, Truerand and Cryptlib are used. In this work, the / dev / random function is used in Ubuntu 16.04 LTS to generate random numbers.

Linux The function / dev / random is described below:

There is a random number generator integrated in the Linux operating system. The source file of this random bit generator, written by Theodore Y. Ts'o [6], is also available. The functions that can be used in the system and their operation are as follows:

• /dev/random: Predicts entropy and returns that much bit.
• /dev/urandom: The PRNG function returns the number of bytes requested by the user.

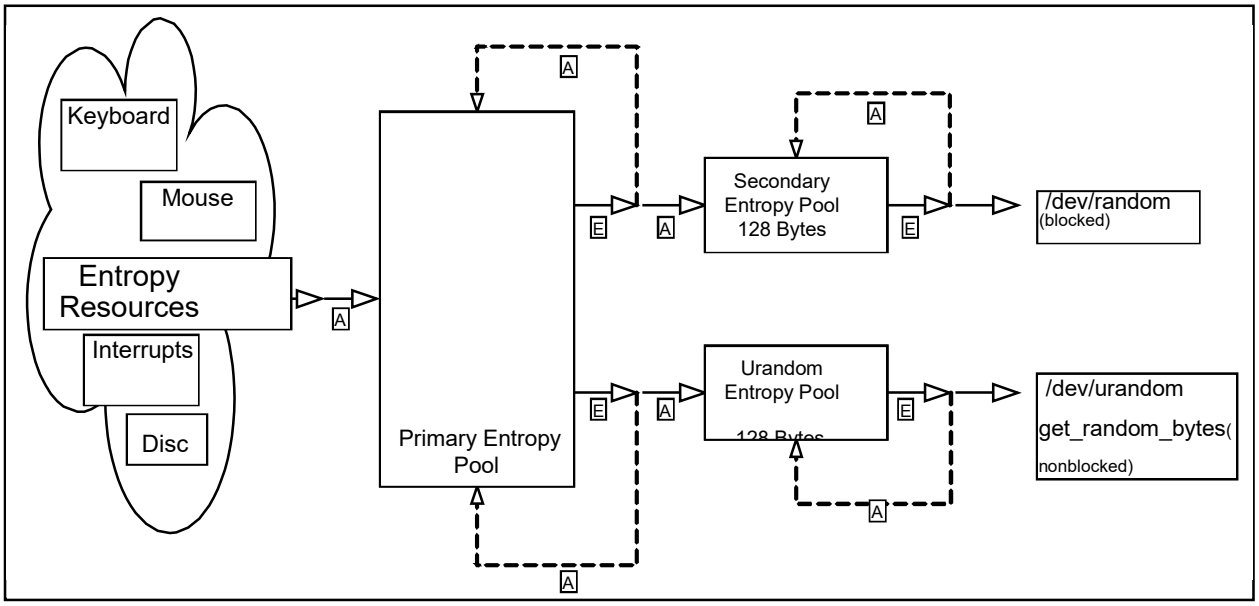

Figure 1 Random Number Generator Working Format in Linux [7]

III. AIS31 STATISTICAL TEST METHOD

Whether or not the resulting bit sequences are random can be determined by applying valid statistical tests. The AIS31 standard consists of 9 tests. Four of these are the same as FIPS 140-2, and the AIS31, NIST 800-22 and FIPS 140-2 tests are shown in Table 1 as comparative.

Table 1 Comparison of AIS31, FIPS140 and SP 800-22 [8]

	AIS.31	FIPS 140	SP 800-22
Purpose	Evaluation criteria for the actual random number generator.	Security requirements that must be met by a cryptographic module.	Randomization of RNG and PRNG is required for cryptographic purposes.

Test	T0: Disjointness test T1: Monobit test T2: Poker test T3: Run test T4: Long Run test T5: Autocorrelation test T6: Üniform distribution test T7: Comparative multinomial test T8: Entropy test	Statistical tests of FIPS 140-1 • Monobit test • Poker test • Run test • Long Run test	15 Statistical tests
Importance level	10^{-6}	10^{-4}	10^{-2}
Relationship	▪ The statistical tests of FIPS 140-1 are the same as T1 to T4 of AIS.31. ▪ The AIS31 is similar to the 800-22, but the significance level of 800-22 is stronger than AIS31. ▪ For the autonomy of the evaluation group, the statistical test was deleted in FIPS 140-2.		

The AIS31 test package consists of three parts: Test 0, Procedure A and Procedure B. Procedure A includes Test 1-2-3-4-5; Procedure B includes Test 6-7-8 tests. Minimum number of bits required for tests;

For Test 0, the bit sequence length must be at least 3145728 bits (2^{16}.48 bits). For Test 1-5, the bit string must have a minimum length of 5140000 bits (20000 * 257 = 5140000 bits). Because 20000 bits are repeated 257 times for each test. (It is usually applied to DRNGs (deterministic RNG), PTRNGs (physical true RNG) and NPTRNGs' (non-physical true RNG) output sequences.)

For the Test 6-7-8, 7200000 bits (depending on the input order, with no fixed value) is required. Generally, binary numbers, das random numbers (binary-valued das-random numbers of PTRNGs) are applied.

A brief description of the tests and their contents in the AIS31 Test Standard is given in Table 2.

Table 2 AIS31 Standard Tests

Procedure A		Test 0, once in 65536 * 48-bit string, repeats the test 257 times from 0 to 5 in successive 20000-bit strings.
Test 0	Disjointness Test	65536 48-bit strings are added, sorted. Two neighbor values should not be equal.
Test 1	Monobit test	The number of 1s should be between 9654 and 10346.
Test 2	Poker test	The 4 bit tuple (ordered sequence value) distribution should be checked for 15 degrees of freedom.
Test 3	Run test	Runs of 1, 2, 3, 4, 5, and 6 ones and zeros are checked for expected occurrences.
Test 4	Long Run test	A single run cannot be bigger than 34.
Test 5	Autocorrelation test	The overlap of the bitstream in the second half of the index is compared to the one with the greatest overlap in the first half of the index.
Procedure B		Distribution tests are performed on 1, 2, 4, 8 bit wide consecutive samples followed by 256000 + 2560 bits of Test 8 test. The total sample size depends on the sample content.
Test 6	Uniform distribution test	Test 6a is a monobit test which allows the number of 1s to be from 25% to 75% of the total. Test 6b is a special Test 7 case with a width of 2.
Test 7	Comparative multinomial test	Less than 10,000 times running distance from the given width is collected and the expected transition probabilities are checked. Test 7a corresponds to width 3, Test 7b corresponds to width 4.
Test 8	Entropy prediction (Coron test)	The nearest precursor distance between the byte values in a 256000 + 2560 bit sequence is accumulated and the empirical entropy is calculated.

IV. AIS31 SOFTWARE AND TEST RESULTS

In order to test the statistical properties of the bit sequences obtained by random number generators, the AIS31 test suite was used. The test documents are examined and the program is written in PHP language. The interface design of the program is shown in Figure 1. The 1mb bit sequence obtained with the command "*dd if=/dev/random of=./testData bs=1M count=1*" was tested through this web application.

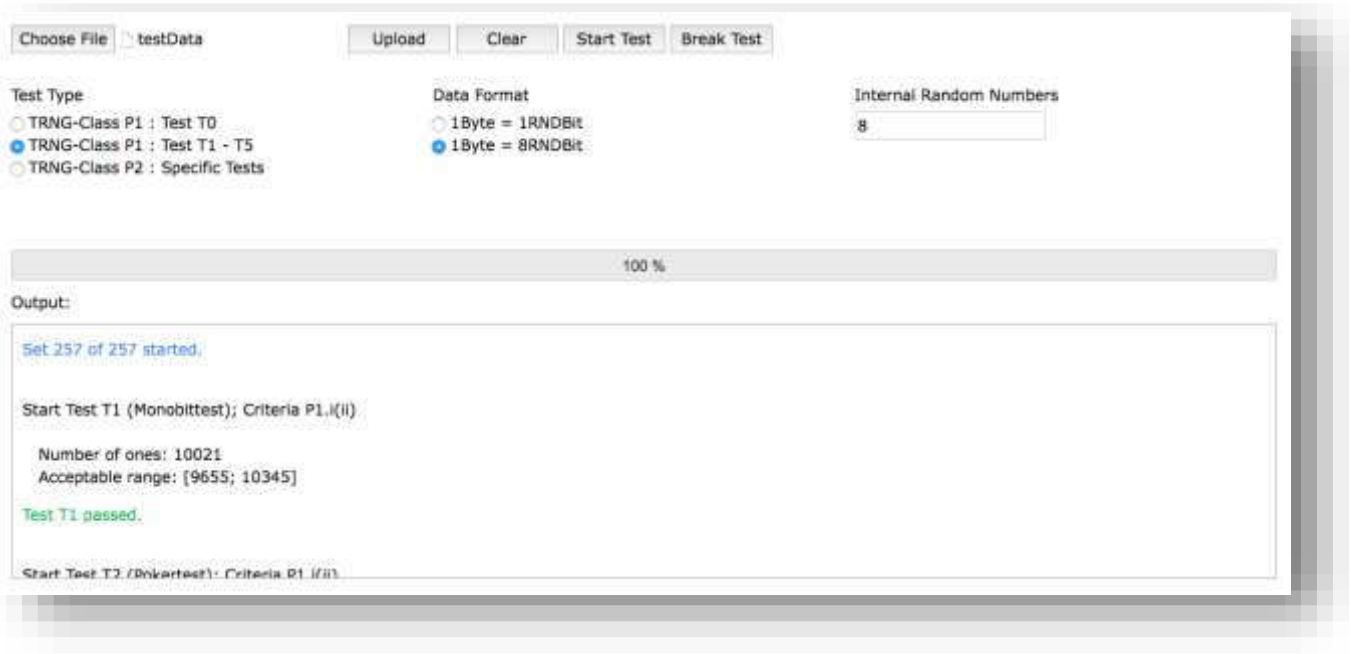

Figure 1 Web Based AIS31 Test Software Interface

In the Linux-based Ubuntu operating system, the data obtained with the random number generator were tested in the web-based AIS31 test software [9] and the results are given in Table 3.

Table 3 AIS 31 Web Based Software Test Results of Random Number Data Produced

Disjointness test	Test 0 Passed (The test is successful if no 48-bit repeated blocks are found in the 65536-bit array.)
Monobit test	Set 257 of 257 started. Start Test T1 (Monobittest); Criteria P1.i(ii) Number of ones: 9862 Acceptable range: [9655; 10345] Test T1 passed.
Poker test	Start Test T2 (Pokertest); Criteria P1.i(ii) Test T2 passed.
Run test	Start Test T3 (Runtest); Criteria P1.i(ii) 0-Runs[1] = 2400; allowed interval: [2267; 2733] 1-Runs[1] = 2534; allowed interval: [2267; 2733] 0-Runs[2] = 1321; allowed interval: [1079; 1421] 1-Runs[2] = 1227; allowed interval: [1079; 1421] 0-Runs[3] = 661; allowed interval: [502; 748] 1-Runs[3] = 641; allowed interval: [502; 748] 0-Runs[4] = 303; allowed interval: [223; 402] 1-Runs[4] = 329; allowed interval: [223; 402] 0-Runs[5] = 169; allowed interval: [90; 223] 1-Runs[5] = 145; allowed interval: [90; 223] 0-Runs[6] = 150; allowed interval: [90; 223] 1-Runs[6] = 128; allowed interval: [90; 223] Test T3 passed.
Long Run test	Start Test T4 (Long Runtest); Criteria P1.i(ii) Test T4 passed.
Autocorrelation test	Start Test T5 (Autocorrelation test); Criteria P1.i(ii) Maximum Z_tau deviation of 2500: 123 Occured for Shifts: Shift: 3714 Repeated Autocorrelation test with Shift: 3714 to Bits 10.000 to 14.999 Z_3714 = 2524 Test T5 passed.
Uniform distribution test	Test procedure T6a for the verification of P2.i) (vii.a). is started. |P(1) - 0.5| = 0.00208 Last Item: 100000 Test T6a passed. Test procedure T6b for the verification of P2.i) (vii.b) is started. p(01) = 0.49818

	p(11) = 0.50387 |p_(01) - p_(11)| = 0.00569 Last Item: 500154 Test T6b passed.
Comparative multinomial test	Test procedure T7a for the verification of P2.i) (vii.c). is started. Last Item: 500154 Test Measurement[0] = 0.0583201047196 Test Measurement[1] = 1.25000460802 Last Item: 1701732 Test T7a passed. Test procedure T7b for verification of P2.i) (vii.d) is started. Test Measurement[0] = 1.63592615697 Test Measurement[1] = 2.20450459257 Test Measurement[2] = 0.250880185551 Test Measurement[3] = 0.192080622341 Last Item: 4912304 Test T7b passed.
Entropy test (Coron test)	Test T8 to verify P2.i) (vii.e). is started. Test Measurement = 8.00126524567 Last Item: 6980784 Test T8 passed.
	Tests completed successfully.

V. Conclusion

The random number sequences obtained by using the Linux function have been subjected to the AIS tests of the web software and the program has worked successfully. The results obtained in Table 3 and Table 4 were successful. The most important feature of the software is that it works in the web environment. Other advantages of the software are that it requires the installation of any additional packages on the computer to be run, that the program interface is user-friendly, and that it is very easy to test the random bit sequence that any non-programming user can easily obtain.

References

[1] L.-Y. Deng and D. K. J. Lin, "Random Number Generation for the New Century," *Am. Stat.*, vol. 54, no. 2, pp. 145–150, 2000.

[2] C. Koc, "Cryptographic Engineering," in *Cryptographic Engineering*, Springer, 2009, pp. 475–504.

[3] BSI, "AIS31 Test Suite," *Bsi*, 2013. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_31_testsuit_zip.zip;jsessionid=E9CA9960CB449465827B2BA22030FBD5.1_cid369?__blob=publicationFile&v=1. [Accessed: 27-Sep-2017].

[4] R. N. Akram, K. Markantonakis, and K. Mayes, "Pseudorandom number generation in smart cards: An implementation, performance and randomness analysis," in *2012 5th International Conference on New Technologies, Mobility and Security - Proceedings of NTMS 2012 Conference and Workshops*, 2012, pp. 1–7.

[5] J. Sobotka and V. Zeman, "Design of the true random numbers generator," *Elektrorevue*, vol. 2, no. 3, pp. 47–52, 2011.

[6] M. Mackall and T. Ts'o, "random. c A strong random number generator, 2009./driver/char/random. c in Linux Kernel 2.6. 30.7." .

[7] Z. Gutterman, B. Pinkas, and T. Reinman, "Analysis of the linux random number generator," in *Proceedings - IEEE Symposium on Security and Privacy*, 2006, vol. 2006, pp. 371–385.

[8] H. Park, J. Kang, and Y. Yeom, "진난수발생기용 난수성 검정 방법 AIS . 31에 대한 확률론적 분석 및 보안성 평가 적용 방법 * Probabilistic Analysis of AIS . 31 Statistical Tests for TRNGs and Their Applications to Security Evaluations *," *J. Korea Inst. Inf. Secur. Cryptol.*, vol. 26, no. 1, pp. 49–67, 2016.

[9] E. Avaroğlu, R. Sinekli, V. Yıldız, and H. Gülcan, "Web Tabanlı AIS31 Test Yazılımı," 2017. [Online]. Available: ais31.mersin.edu.tr. [Accessed: 28-Sep-2017].