THE POWER OF NIST CRYPTOGRAPHIC STATISTICAL TESTS SUITE

EMIL SIMION[*], PAUL BURCIU[**]

[*] University Politehnica of Bucharest, Bucharest, Romania
[**] Military Technical Academy, Bucharest, Romania

Corresponding author: EMIL SIMION, E-mail: emil.simion@upb.ro

Abstract. This paper is focused on an open question regarding the correlation and the power of NIST statistical test suite. If we found some correlation between these statistical tests, then we can improve the testing strategy by executing only one of the tests that are correlated. Using the Galton-Pearson "product-moment correlation coefficient", by simulation, we found a high correlation between five couples of these statistical tests. Also we make a conjecture about the power of NIST statistical tests suite in the case that these tests are independent.

Key words: Statistical testing, Cryptographic evaluation, Random bit generators.

1. INTRODUCTION

Kerckhoffs's principle states that a cryptographic primitive will be secure if everything about it, except the key, is known. To evaluate the security of cryptographic primitives, we have at our disposal analytical tools and statistical-probabilistic tools. The latter, unlike the former, are universally valid and can be applied to any cryptographic primitive. Statistical tests are an efficient tool for assigning the ownership of a set of independent observations, called measurements, to a specific population or probability distribution; they are commonly used in the field of randomness testing. Measurements are the first step that leads to control and improve the random behaviour of a true random bit generator (and others cryptographic functions). If we do not measure the degree of randomness we cannot understand and control the behaviour of the device (or algorithm), thus we cannot improve its security. If we analyse the output of the cryptographic primitive and find non-uniform patterns, then it can be possible to break it. But if we do not find these non-uniform patterns, no one can guarantee that there will be no analytical methods for breaking it. This paper is an extension of our previous works, [1] and [2], regarding the independence of statistical test suite NIST SP 800-22 and it is organized as follows.

In section 2 of this paper we present statistical requirements for validating the security of cryptographic primitives. Validation by statistical methods is prone to errors due to the samples used in testing. In section 3 we discuss types of errors, sample requirements, and constructions for testing block ciphers. The statistical tools and methods used in security evaluation of the block cipher are generally based on the "de facto" standard STS SP 800-22 [3], a publication of Computer Security Research Center, a division of NIST, that initially described sixteen statistical tests (because of improper evaluation of mean and variance, the Lempel-Ziv test was dropped from the revised version). Therefore, we discuss about STS SP 800-22 and the statistical cryptographic evaluation standard used in AES candidates' evaluation [4] and provide some important aspects regarding the independence of test suite. Also we give an important result regarding the power of NIST statistical test (probability to reject a false hypothesis) in the case that these tests are independent: the distribution of the sum of all *P*-values is a normal one! In section 4 we provide experimental results regarding evaluation of correlation between statistical tests that were run using three different lengths of the string sample (i.e. 1 up to 6 million bits). In fact, when using the Galton-Pearson "product-moment correlation coefficient" we found a high correlation between some couples of these statistical tests. This fact allows us to improve the testing strategy by executing only the uncorrelated statistical tests. Finally, in section 5, we conclude.

2. THE RELEVANCE OF STATISTICAL TESTING IN CRIPTO VALIDATION

As we stated in [1] and [2], when designing cryptographic primitives, such as block/stream ciphers, there are several requirements. One of these requirements is that the cryptographic primitive has to satisfy several statistical properties:

- strict avalanche: changing one input bit causes on average about 50% output changes;
- correlation immunity: correlated input gives an uncorrelated output;
- predictability: having a sample of n binary observations it is impossible to predict (with a different from 0.5 probability) the next bit outcome;
- balance: every output is produced by the same number of inputs.

The validation of these criteria is done by analytical methods or statistical tests (in case the first one is not available). Also, statistical tests are useful to mount distinguishing attacks that allow an attacker to distinguish random data from encrypted data.

Statistical hypothesis testing is a mathematical technique, based on sample data, used for supporting the decision making on the theoretical distribution of a population. In the case of statistical analysis of a cryptographic algorithm, the sample is the output of the algorithm from different inputs for the key and plain text. Because we deal with sample data from the population, the decision process of the population's probability distribution is prone to errors. To meet this challenge, we model the decision making-process with the aid of two statistical hypotheses: the null hypothesis, denoted by H_0 - in this case, the sample does not indicate any deviation from the theoretical distribution - and the alternative hypothesis H_A - when the sample indicates a deviation from the theoretical distribution.

There can be two types of errors: first type error (also known as the level of significance), i.e. the probability of rejecting the null hypothesis when it is true:

$$\alpha = \Pr\left(\text{reject } H_0 \middle| H_0 \text{ is true}\right) \tag{1}$$

and the second type error, which represents the probability of failing to reject the null hypothesis when it is false:

$$\beta = \Pr\left(\text{accept } H_0 \middle| H_0 \text{ is false}\right) \tag{2}$$

These two errors, α and β, cannot be minimized simultaneously since the risk β increases as the risk α decreases and vice-versa. For this reason, one solution is to have the value of α under control and compute the probability β.

The analysis plan of the statistical test includes decision rules for rejecting the null hypothesis. These rules can be described in two ways:

- decision based on P-value. In this case, we consider f to be the value of the test function and compare the P-value, defined as $\Pr(X < f)$, with the value α, and decide on the null hypothesis if P-value is greater than α;
- the "critical region" of a statistical test is the set which causes the null hypothesis to be rejected; the complementary set is called the "acceptance region". In the acceptance region, we shall find the ideal results of the statistical test.

Because for each statistical test the rejection rate α is a probability, which is "approximated" from the sample data, we need to compute the minimum sample size in order to achieve the desired rejection rate α. Also, the sample must be independent and governed by the same distribution.

A way to construct samples for testing block ciphers is to setup the plain text and the key: $X_i=E(P_i;k_i)$ where E is the encryption function, P_i is the set of plain texts, and k_i is the set of keys. For each plain text input P_i and each encryption key k_i, the output from the encryption function must have a uniform distribution. To test this assumption, for AES candidates, Soto [4] constructed the samples with low/high density plain text/key (a low density text/key is a text/key with a small number of 1s, in opposition to a high density text/key which is a text/key with a small number of 0s). As we can see, when using this type of construction, the samples are not independent variables because they are connected by means of the encryption function E. Are the results of the statistical tests relevant when this assumption is not true? If the statistical test accepts the null hypothesis, then we can say that there is not enough evidence for the non-uniformity of the sample.

If a cryptographic primitive passed a statistical test, it does not mean that the primitive is secure. For example, the predictable sequence 01010…01 is "perfect" if we analyzed it with the bit frequency test. This is one of the reasons why we should be "suspicious" if we obtained perfect results. To avoid these situations, in some cases it is indicated to include the neighbourhood of the ideal result in the critical region.

NIST SP 800-90A contains the specifications of four cryptographic secure PRBG for use in cryptography based on: hash functions, hash-based message authentication code, block ciphers and elliptic curve cryptography. Some problems with the later one (Dual_EC_DRBG) were discovered since 2006 [5]: the random numbers it produces have a small bias and it raises the question if NSA put a secret backdoor in Dual_EC_DRBG. It was proved, in 2013, that (Dual_EC_DRBG) has flows. To restore the confidence on encryption standards, NIST reopens the public vetting process for the NIST SP 800-90A.Thus, if the algorithm failed to certain tests, then it should not be used in cryptographic applications because an attacker might be able to predict the behaviour of the algorithm or, even worse, may indicate the existence of certain trapdoors.

3. THE POWER OF STATISTICAL TEST SUITE SP 800-22

Because STS SP 800-22 is a standard, we shall focus on it rather than other statistical test suites. STS SP 800-22 (the revised version) consists of fifteen statistical tests, which highlight a certain fault type proper to randomness deviations. Each test is based on a computed test statistic value f, which is a function of the sample. A statistical test is used to compute a P-value=$\Pr(f|H_0)$ that summarizes the strength of the evidence against the null hypothesis. If the P-value is greater, then the null hypothesis is accepted (the sequence appears to be random). The tests are not jointly independent, making it difficult to compute an overall rejection rate (i.e. the power of the test). Recall that the tests $T_1,\ldots,T_{15}$ will be jointly independent if (3)

$$\Pr(T_{i1},\ldots,T_{ik}) = \Pr(T_{i1})\cdot\ldots\cdot\Pr(T_{ik}) \tag{3}$$

is true for every subset $\{i_1,\ldots,i_k\}$ of $\{1,\ldots,15\}$. Obviously, jointly independent tests will be pair wise independent. The converse is not true. If the statistical tests were independent, then the overall rejection rate would be computed using the probability of the complementary event given by (4):

$$1-(1-\alpha)^{15} \approx 0.14 \tag{4}$$

In Table 1 we summarize the reference distribution of NIST statistical tests:

Table 1. Reference distributions of NIST statistical tests

Test	Reference distribution
Frequency (Monobit) test	half normal
Frequency Test within a Block	$\chi^2(N)$
Runs Test	normal
Test for the Longest Run of Ones in a Block	$\chi^2(K)$
Binary Matrix Rank Test	$\chi^2(2)$
Discrete Fourier Transform (Spectral) Test	normal
Non-overlapping Template Matching Test	$\chi^2(N)$
Overlapping Template Matching Test	$\chi^2(K)$
Maurer's "Universal Statistical" Test	normal
Linear Complexity Test	$\chi^2(K)$
Serial Test	$\chi^2(2^{m-1})+\chi^2(2^{m-2})$
Approximate Entropy Test	$\chi^2(2^m)$
Cumulative Sums (Cusum) Test	normal
Random Excursions Test	$\chi^2(5)$
Random Excursions Variant Test	half normal

Thus, we conjecture that if we sum all the *P*-values of the statistical tests we shall obtain a normal distribution. As we can see, there are three types of distributions: normal, half normal and χ^2.

It is well known that the sum between two independent variables χ^2(n) and χ^2(m) is χ^2(n + m) variable. If we assume that all the χ^2 distributions are independent and compute the sum of all corresponding *P*-values of these distributions, we shall obtain a χ^2 distribution with the number of freedom degrees greater then 30, which is well approximated by the normal distribution. Also it is known that the sum between two independent normal variables it is a normal random variable.

Thus, if we assume that the normal and χ^2 statistical tests are independent, then the sum of all *P*-values of these tests will go after the normal distribution. The result is similar to the interpretation of tomographic images which are aggregations of all taken images. Thus we may cumulate almost (except frequency test and random excursions test) statistical tests into only one test.

STS SP 800-22 provides two methods for integrating the results of the tests, namely percentage of passed tests and the uniformity of *P*-values. The experiments revealed that these decision rules were insufficient and, therefore, researchers considered their improvement would be useful. Therefore, in [6], new integration methods for these tests were introduced:

- maximum value decision, based on the max value of independent statistical test T_i, i=1,…,n. In this case, the maximum value of the random variables was computed; the repartition function of the max value, given by (5):

$$\Pr\left(\max(T_1,...,T_n) < x\right) \tag{5}$$

this being the product of the repartition functions of the random variables given by (6):

$$\mathrm{T_i} : \prod_{i=1}^{n} \Pr(T_i < x) \tag{6}$$

- sum of square decision, based on the sum of squares S of the results of the tests (which have a normal distribution). The distribution of *S*, in this case, is χ^2, the freedom degrees given by the number of partial results which are being integrated.

Weak points of STS SP 800-22 are:

- fixed first order error α=0.01;
- the tests are not evaluating the second order error, which represents the probability to accept a false hypothesis.

In [7], the possibility of extending STS SP 800-22 tests to arbitrary level of significance α (and computing β) is presented by computing, for n>30, the second order probability:

$$\beta = \Phi\left(\sqrt{\frac{p_0 q_0}{p_1 q_1}}\left(u_{1-\frac{\alpha}{2}} - \frac{n(p_1 - p_0)}{\sqrt{np_0 q_0}}\right)\right) - \Phi\left(\sqrt{\frac{p_0 q_0}{p_1 q_1}}\left(u_{\frac{\alpha}{2}} - \frac{n(p_1 - p_0)}{\sqrt{np_0 q_0}}\right)\right) \tag{7}$$

In [8], there are some comments about NIST statistical testing methodology: ambiguous hypothesis (does not specify the family of distribution and/or the alternative), error quantification (NIST does not give the size of the category-test decisions), power of the test suite, dependencies of tests, invariant test (cryptographically equivalent tests performed on the same sample do not necessary give the same result), and inadmissible tests (the existence of better tests).

After the process of evaluation of AES candidates, several academic studies reported that the test setting of Discrete Fourier Transform test (designed to detect periodic features in the tested sequence that would indicate a deviation from the assumption of randomness) and Lempel-Ziv test (designed to see if the sequence can be compressed and will be considered to be non-random if it can be significantly compressed) of the STS SP 800-22 are unsuitable:

- threshold value and the variance σ^2 of theoretical distribution, and
- the setting of standard distribution, which has no algorithm dependence (SHA-1 for million bit sequences) and the re-definition of the uniformity of *P*-values (based on simulation).

Because the mean and variance of Lempel-Ziv test were evaluated using samples generated by an algorithm, in the revised version of STS SP 800-22 the Lempel-Ziv was dropped.

4. EXPERIMENTAL ANALYSIS OF CORRELATION BETWEEN STATISTICAL TESTS

In [7], we studied the variation of a second order error β, with respect to p_1 and a length n of bit stream Frequency test within a block, Runs, Discrete Fourier transform (spectral), and Serial test (2 components). For the rest of statistical tests, it is difficult to find an analytical formula for the second order error β. For this reason, a proposal is the following procedure for checking the independence of tests i and j:

- implement the NIST SP 800-22 testing suite;
- use a "good" pseudorandom generator GPA to test N binary samples;
- for each test i, define the Bernoulli random variable T_i which will give 1 if the sample passes the test, otherwise 0;
- estimate the value of $\Pr(T_i \text{ and } T_j) - \Pr(T_i)\Pr(T_j)$. If the tests are independent, then this value should be close to zero.
- find the highest value of the above value for i and j.

On the other hand, the result of a statistical test, denoted as P-value, as a measure of randomness, ranges between [0,1] and is calculated by a specific formula given for each test by NIST's specification. With a P-value close to 1, we have a high level of randomness.

Our work improves the results of [9] and [10] and, based on the Galton-Pearson "product-moment correlation coefficient" ([11]), evaluates pairs of P-values and produces a result which ranges between [-1, 1]. A correlation of +1 means that there is a perfect positive linear relationship between variables, or a direct proportion, while a correlation of -1 means that there is a perfect negative linear relationship between them, or an inverse proportion. With a correlation which is close to the absolute value of 1, we have a strong relationship between the variables. In case of a correlation close to 0, the variables are independent. The reciprocal is not always true ([11]).

In order to evaluate any specific correlation between results of statistical tests and to produce reliable/effective results and conclusions, this analysis was done by calculating and analyzing 6 sets of coefficients obtained by applying NIST statistical tests to 100 binary samples of different ascending length (i.e. 1 up to 6 million bits).

The number of samples was chosen according to NIST's specifications where the unique value of minimum 200, that is, for Linear Complexity Test, was intentionally not accomplished due to the fact that this test works with a fixed number of 500 substrings. Hence, the limit in fact was accomplished.

As requested by NIST's specifications, a pseudorandom binary string of approximately 1 billion bits was generated by using an FPGA loop implementation of the encryption part of a well-known symmetrical cryptographic algorithm, AES-128 [13], with a "1h" (i.e. hexadecimal value) unique key. On this loop, every encrypted output binary sequence was taken and applied as an input to the subsequent encryption. Knowing that for a single encryption simulation, that is, for 128 bits, it took 240 ns, a 1.875 s simulation (or 7,812,500 iterations) was needed to be run, in order to produce the 1-billion-bits binary string. This simulation was done by using Xilinx ISE Design Suite (shareware version 14.7) and one of the authors' previous hardware implementations of AES-128 [14].

With the intention to make experiments practical/efficient, the NIST test suite (version 2.1.2) was implemented according to NIST's specifications on five Linux OS (Ubuntu version 18.04.1 Desktop 64-bit) virtual machines (4 processors, 4 GB of RAM), all running on a single physical desktop PC (Intel I7 Quad Core, 16 GB of RAM). The virtual machines were created with the VMWare Workstation (shareware version 12.5.5) software.

All 15 tests were used, 3 of them being treated like double tests as follows:

- cumulative Test, denoted as T3, consists of Forward (T3F) and Reverse (T3R) tests;
- non Periodic Template Matchings Test, denoted as T8, was approached as for 2 binary sequences, "000000001" (T8.1) and "111111110" (T8.2), respectively;
- serial Test, denoted as T14, was treated like 2 tests (T14.1 and T14.2) corresponding to 2 P-values produced by this test.

Therefore, instead of 15, 18 individual tests were considered, being listed by Table 2.

Table 2. Used tests list

Test Number	Test Name	Test Variants
T1	Frequency (Monobit)	
T2	Frequency Test within a Block	
T3	Cumulative Sums (Cusums)	T3F; T3R
T4	Runs	
T5	Longest-Run-of-Ones in a Block	
T6	Binary Matrix Rank	
T7	Discrete Fourier Transform (Spectral)	
T8	Non-overlapping Template Matching	T8.1; T8.2
T9	Overlapping Template Matching	
T10	Maurer's “Universal Statistical”	
T11	Approximate Entropy	
T12	Random Excursions	
T13	Random Excursions Variant	
T14	Serial	T14.1; T14.2
T15	Linear Complexity	

As mentioned before, the choice was to use samples of 6 different ascending lengths in order to check any possible dependence between correlation coefficients and the sample length. Moreover, the option of using 100, as a unique number of samples, was motivated by the necessity of having uniform results, such that they could be compared. In case this requirement was not complied with, this comparison would not be possible.

In order to relieve certain correlations between the results of statistical tests and to give reliable/effective conclusions, only correlation coefficients greater than or equal to 0.5 (similarly to [10]) were taken into consideration, avoiding to set too high limits and to neglect any dependencies with lower coefficients that might occur.

For evaluation of correlations between statistical test results, the chosen method was Galton-Pearson formula, that is, the correlation coefficient. In order to produce reliable/effective results and conclusions, this was done by calculating and analyzing six consecutive sets of correlation coefficients, corresponding to applying NIST statistical tests over 100 binary samples of different lengths (i.e. 1 up to 6 million bits).

The correlation coefficients that resulted from applying NIST statistical tests, showing a strong correlation (close to or greater than 0.5) between a test situated on the horizontal and one on the vertical line, are contained by Table 3 - 8 shown below (only tests with correlations), with sample length denoted as M between 1,000,000 and 6,000,000 bits.

Table 3. Correlation coefficients for M = 1,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0.738	0.722	0.287	0.248	0.031	-0.002
T3F	0.738	1	0.765	0.371	0.313	-0.087	-0.245
T3R	0.722	0.765	1	0.235	0.180	-0.049	-0.149
T12	0.287	0.371	0.235	1	0.725	-0.010	-0.037
T13	0.248	0.313	0.180	0.725	1	-0.011	-0.079
T14.1	0.031	-0.087	-0.049	-0.010	-0.011	1	0.690
T14.2	-0.002	-0.245	-0.149	-0.037	-0.079	0.690	1

Table 4. Correlation coefficients for M = 2,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0.790	0.767	0.286	0.324	0.022	-0.052
T3F	0.790	1	0.705	0.421	0.348	-0.092	-0.116
T3R	0.767	0.705	1	0.236	0.201	-0.043	0.033
T12	0.286	0.421	0.236	1	0.623	0.128	0.036
T13	0.324	0.348	0.201	0.623	1	0.049	-0.098
T14.1	0.022	-0.092	-0.043	0.128	0.049	1	0.690
T14.2	-0.052	-0.116	0.033	0.036	-0.098	0.690	1

Table 5. Correlation coefficients for M = 3,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0,775	0,797	0,161	0,144	-0,086	0,042
T3F	0,775	1	0,793	0,287	0,274	-0,054	-0,010
T3R	0,797	0,793	1	0,160	0,148	0,071	0,070
T12	0,161	0,287	0,160	1	0,572	-0,051	0,017
T13	0,144	0,274	0,148	0,572	1	-0,173	-0,151
T14.1	-0,086	-0,054	0,071	-0,051	-0,173	1	0,752
T14.2	0,042	-0,010	0,070	0,017	-0,151	0,752	1

Table 6. Correlation coefficients for M = 4,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0,743	0,807	0,213	0,315	0,143	0,202
T3F	0,743	1	0,775	0,235	0,345	-0,008	0,076
T3R	0,807	0,775	1	0,212	0,282	0,044	0,167
T12	0,213	0,235	0,212	1	0,592	0,030	-0,011
T13	0,315	0,345	0,282	0,592	1	-0,079	-0,060
T14.1	0,143	-0,008	0,044	0,030	-0,079	1	0,687
T14.2	0,202	0,076	0,167	-0,011	-0,060	0,687	1

Table 7. Correlation coefficients for M = 5,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0.716	0.733	0.199	0.139	-0.123	-0.111
T3F	0.716	1	0.637	0.267	0.099	-0.107	-0.117
T3R	0.733	0.637	1	0.086	0.014	-0.164	-0.106
T12	0.199	0.267	0.086	1	0.498	-0.056	-0.135
T13	0.139	0.099	0.014	0.498	1	-0.013	-0.023
T14.1	-0.123	-0.107	-0.164	-0.056	-0.013	1	0.746
T14.2	-0.111	-0.117	-0.106	-0.135	-0.023	0.746	1

Table 8. Correlation coefficients for M = 6,000,000 bits

Tests	T1	T3F	T3R	T12	T13	T14.1	T14.2
T1	1	0.745	0.753	0.193	0.372	0.03	-0.018
T3F	0.745	1	0.711	0.166	0.329	0.089	0.08
T3R	0.753	0.711	1	0.003	0.226	0.085	0.084
T12	0.193	0.166	0.003	1	0.474	-0.08	-0.119
T13	0.372	0.329	0.226	0.474	1	0.03	-0.007
T14.1	0.03	0.089	0.085	-0.08	0.03	1	0.679
T14.2	-0.018	0.08	0.084	-0.119	-0.007	0.679	1

where: T1 - Frequency (Monobit), T3F - Cumulative Sums (Forward), T3R - Cumulative Sums (Reverse), T12 - Random Excursions, T13 - Random Excursions Variant, T14.1 - Serial 1 (where a P-value$_1$ was evaluated for $K_1 = 2^{m-1}$ degrees of freedom, with m being the number of bits in a pattern that appears in the n-bit stream), and T14.2 - Serial 2 (where a P-value$_2$ was evaluated for $K_2 = 2^{m-2}$ degrees of freedom); the values that are close to or greater than 0.5 were filled with grey color.

We found a high correlation between five couples of these statistical tests: (frequency, cumulative sums Forward), (frequency, cumulative sums reverse), (cumulative sums forward, cumulative sums reverse), (random excursions, random excursions variant) and (serial 1, serial 2). This allows us to improve the testing strategy by "dropping" one of the correlated tests.

Looking at the correlation coefficients, concerning only presumed dependencies (correlations), we found different patterns of variation (depending on the sample length), as follows:

T1-T3F: 0.738 ↗ 0.790 ↘ 0,775 ↘ 0,743 ↘ 0.716 ↗ 0.745 - Large Oscillation pattern

T1-T3R: 0.722 ↗ 0.767 ↗ 0,797 ↗ 0,807 ↘ 0.733 ↗ 0.753 - Large Oscillation pattern

T3F-T3R: 0.765 ↗ 0.705 ↗ 0,793 ↘ 0,775 ↘ 0.637 ↗ 0.711 - Small Oscillation Pattern

T12-T13: 0.725 ↘ 0.623 ↘ 0,572 ↗ 0,592 ↘ 0.498 ↘ 0.474 - Small Oscillation Pattern

T14.1-T14.2: 0.690 ↗ 0.690 ↗0,752 ↘ 0,687 ↗ 0.746 ↘ 0.679 - Small Oscillation Pattern

These patterns will be object of our future work in order to mathematically describe the variance of correlation coefficients with the length of string sample.

5. CONCLUSIONS

In this article we focused on an open question regarding the correlation of the NIST statistical test suite and improved the results obtained in [7], [9], and [10]. Using the Galton-Pearson "product-moment correlation coefficient" we found a high correlation between five couples of these statistical tests. This allowed us to improve the testing strategy. Also we make a conjecture about the power of NIST statistical teste suite in the case that these tests are independent.

6. REFERENCES

1. E. Simion, P. Burciu, *A Note On the Correlations Between NIST Cryptographic Statistical Tests Suite*, U.P.B. Scientific Bulletin, Series A, **Vol. 81**, Issue. 1, 2019, ISSN 1223-7027.
2. E. Simion, P. Burciu, *A Systematic Approach of NIST Statistical Tests Dependencies*, Journal of Electrical Engineering, Electronics, Control and Computer Science - JEEECCS, **Vol. 5**, Issue 15, pages 1-6, 2019.
3. NIST SP 800-22 Revision 1a, *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*, https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf.
4. NIST standards: http://www.nist.gov/, http://www.csrc.nist.gov/. [Soto] J. Soto, *Randomness Testing of the Advanced Encryption Standard Candidate Algorithms*, NIST IR 6390, September 1999.

5. D. R. L. Brown and K. Gjosteen, *A Security Analysis of the NIST SP 800-90 Elliptic Curve Random Number Generator*, Cryptology ePrint Archive, Report 2007/048.
6. A. Oprina, A. Popescu, E. Simion, and Gh. Simion, *Walsh-Hadamard Randomness Test and New Methods of Test Results Integratio*n, Bulletin of Transilvania University of Braşov, **vol. 2(51)** Series III-2009, pg. 93-106.
7. C. Georgescu, E. Simion, *New results concerning the power of NIST randomness tests*, Proceedings of the Romanian Academy Series A, **Vol. 18**, 2017.
8. S. Murphy, *The power of NIST's statistical testing of AES candidates*, Preprint. January 17, 2000.
9. J. Kelsey, K.A. McKa, M. Sönmez Turan, *Predictive Models for Min-entropy Estimation*. In: Güneysu T., Handschuh H. (eds) Cryptographic Hardware and Embedded Systems -- CHES 2015. CHES 2015. Lecture Notes in Computer Science, vol. 9293. Springer, Berlin, Heidelberg.
10. A. Doğanaksoy, F. Sulak, M. Uğuz, O. Şeker, and Z. Akcengiz: *Mutual Correlation of NIST Statistical Randomness Tests and Comparison of Their Sensitivities on Transformed Sequences*, Turkish Journal of Electrical Engineering & Computer Sciences, Turkey, 2017.
11. J. L. Rodgers, W. A. Nicewander, *Thirteen Ways to Look at the Correlation Coefficient*, The American Statistician, **Vol. 42**, No. 1, Feb., 1988.
12. D. S. Moore, W. I. Notz, M. A. Fligner, *The Basic Practice of Statistics - 3rd edition*, W. H. Freeman & Co., New York, NY, USA, 2003.
13. NIST: FIPS PUB 197, Announcing the Advanced Encryption Standard (AES), 2001.
14. P. Burciu: Design and Optimization Methods for Hardware Implementation of Information Enciphering Algorithms on Digital Communications, PhD Thesis, University of Pitesti, Pitesti, Romania, 2009.