A NOTE ON THE CORRELATIONS BETWEEN NIST CRYPTOGRAPHIC STATISTICAL TESTS SUITE

Emil SIMION[1], Paul BURCIU[2]

This paper is focused on an open question regarding the correlation and the power of the 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 this 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).

Keywords: statistical testing, cryptographic evaluation, random bit generators.

1. Introduction

When we talk about communications security, we need to cover both the confidentiality of the transmitted data and the confidentiality of the communicators (sender and receiver). 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 cryptography, specifically in randomness testing. Statistics can be useful in showing that a proposed system is weak. Thus, one criterion in validating ciphers is that there is no efficient method for breaking it by brute force. That is, if we have a collection of cipher texts (and eventually the corresponding plain texts) all the keys have the same probability to be the correct key, thus we have uniformity in the key space. If we are analyzing the output of the cipher and find non-uniform patterns, then it can be possible to break it. But if we cannot find these non-uniform patterns, no one can guarantee that there are no analytical methods for breaking it. Also, statistical tests can be used for analyzing communication data and detect covert communications (steganographic systems) and anomalies in TCP flow (cyber-attacks).

The paper will be organized as follows. In section 2 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

[1] Lect., Dept. of Mathematical Methods and Models, University POLITEHNICA of Bucharest, Romania, e-mail: emil.simion@upb.ro
[2] PhD Eng., Military Technical Academy of Bucharest, Romania, e-mail: pburciu@yahoo.com

3 we discuss types of errors, sample requirements, and constructions for testing block ciphers. For a reference to clearly defined security, the International Standardization Organization (ISO), national standards organizations such as American National Standards Institute (ANSI), National Institute for Standards and Technologies (NIST) standardize requirements and evaluation criteria for cryptographic algorithms. The statistical methods used in academic security evaluation of the AES candidates are generally based on the "de facto" standard STS SP 800-22 [8], a publication of Computer Security Research Center [9], a division of NIST, that initially describes sixteen statistical (because improper evaluation of mean and variance, the Lempel-Ziv test was dropped from the revised version). Besides the above, there exist other several statistical testing procedures and tools specified in Donald Knuth's book [3], The Art of Computer Programming, Seminumerical Algorithms, the Crypt-XS suite of statistical tests developed by researchers from the Information Security Research Centre at Queensland University of Technology from Australia, the DIEHARD suite of statistical tests developed by George Marsaglia [5], TestU01, a C library for empirical testing of random number generators developed by P. L'Ecuyer and R. Simard [4]. In section 3 we discuss about STS SP 800-22 and the statistical cryptographic evaluation standard used in AES candidates' evaluation. 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, 2, and 5 million bits). In fact, 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. Statistical Testing of Cryptographic Primitives

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 (1):

$$\alpha = Pr(\text{reject } H_0 \mid H_0 \text{ is true}) \tag{1}$$

and the second type error, which represents the probability of failing to reject the null hypothesis when it is false (2):

$$\beta = Pr(\text{accept } H_0 \mid H_0 \text{ is false}) \tag{2}$$

These two errors, α and β, can't 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 (3):
$$Pr(X < f) \tag{3}$$
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 test 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 (4):

$$X_i = E(P_i, k_i) \tag{4}$$

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, in NIST standard [9] the samples are constructed 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 passes a statistical test, it does not mean that the primitive is secure. For example, the predictable sequence 01010…01 is "perfect" if we analyze it with the bit frequency test. This is one of the reasons why we should be "suspicious" if we obtain perfect results. To avoid these situations, in some cases it is indicated to include the neighborhood of the ideal result in the critical region.

NIST SP 800-90A [NIST SP 800-90] 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 ([2]): 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. Internal memos leaked by a former NSA contractor, Edward Snowden, suggest that NSA generated a trapdoor in Dual_EC_DRBG. To restore the confidence on encryption standards, NIST reopens the public vetting process for the NIST SP 800-90A.Thus, if algorithm will fail to certain tests, then it should not be used in cryptographic applications because an attacker will be able to predict the behavior of the algorithm or, even worse, may indicate the existence of certain trapdoors.

3. A View on STS SP 800-22

Pseudorandom bit generators (PRBG) are cryptographically secure if pass *next bit test*, that is, there is no polynomial time algorithm which, given the first l-bits of the output, can predict $l+1$-bit with probability significantly greater than 0.5, and in the situation when a part of PRBG is compromised, then it should be impossible to reconstruct the stream of random bits prior to the compromising. Yao [Yao] proved that PRBG passes next bit test if and only if passes all polynomial time statistical tests. Because practically is not feasible to test PRBG

for all polynomial statically tests, we need to find a representative, polynomial time, statistical testing suite such as STS SP 800-22.

Because STS SP 800-22 is a standard, we shall focus on it rather than other statistical test suites ([3], [4], or [5]). 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 (5):

$$P-Value = Pr(f \mid H_0) \tag{5}$$

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 $T_1,\ldots,T_{15}$ are jointly independent if (6) is true for every subset $\{i_1,\ldots,i_k\}$ of $\{1,\ldots,15\}$:

$$Pr(T_{i1},...,T_{ik}) = Pr(T_{i1}) \cdots Pr(T_{ik}) \tag{6}$$

Obviously, jointly independent tests are pair wise independent. The converse is not true [1]. If the statistical tests would be independent, then the overall rejection rate, would be computed using the probability of the complementary event (7):

$$1-(1-\alpha)^{15} \approx 0.14 \tag{7}$$

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 [10], 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 being the product of the repartition functions of the random variables T_i (8):

$$Pr(max(T_1,...,T_n) < x) = \prod_{i=1}^{n} Pr(T_i < x) \tag{8}$$

- 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:

- Fixed first order error $\alpha = 0.01$;
- 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 (9):

$$\beta = \Phi\left(\sqrt{\frac{p_0 q_0}{p_1 q_1}}\left(u_{1-\frac{\alpha}{2}} - \frac{n(p_1 - p_0)}{\sqrt{np_0 q_0}}\right)\right) - \Phi\left(\sqrt{\frac{p_0 q_0}{p_1 q_1}}\left(u_{\frac{\alpha}{2}} - \frac{n(p_1 - p_0)}{\sqrt{np_0 q_0}}\right)\right) \quad (9)$$

In [6], 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, researchers [Kim] 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 [10], we studied the variation of the second order error β, with respect to p_1 and the length n of the 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, one proposal is the following procedure for checking the independence of tests i and j:

i) implement the NIST SP 800-22 testing suite;
ii) use a “good” pseudorandom generator GPA to test N binary samples;
iii) for each test i, define the Bernoulli random variable T_i which gives 1 if the sample passes the test, otherwise 0;

iv) estimate the value of (10):

$$Pr(T_i \text{ and } T_j) - Pr(T_i) \cdot Pr(T_j) \quad (10)$$

If the tests are independent, then this value should be close to zero.

v) 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 [11] and [12] and, based on the Galton-Pearson "product-moment correlation coefficient" ([13]), 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 ([14]). For the evaluation of correlation 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 three sets of correlation coefficients, corresponding to the application of NIST statistical tests over 100 binary samples of different lengths (i.e. 1, 2, and 5 million bits). The correlation coefficients that resulted from the application of NIST statistical tests, and showed a strong correlation (close to or greater than 0.5) between a test situated on the horizontal line and one on the vertical line, are contained by Table 1, 2, and 3 shown below (only the tests with correlations), that is, for a sample length M = 1,000,000, 2,000,000, and 5,000,000 bits.

Table 1

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 2

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 3

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

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 the presumed dependencies (correlations), we found different patterns of variation (depending on the sample length), as follows:

- Oscillation pattern: T1-T3F: 0.738 ↑ 0.790 ↓ 0.716
- Oscillation pattern: T1-T3R: 0.722 ↑ 0.767 ↓ 0.733
- Decrease pattern: T3F-T3R: 0.765 ↓ 0.705 ↓ 0.637
- Decrease pattern: T12-T13: 0.725 ↓ 0.623 ↓ 0.498
- Increase pattern: T14.1-T14.2: 0.690 ↑ 0.690 ↑ 0.746

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.

6. 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 [10], [11] and [12]. 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.

R E F E R E N C E S

[1] *Sergei Natanovich Bernstein*, Theory of Probability, 4th ed. (in Russian), Gostechizdat, Moscow-Leningrad, 1946.
[2] *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.
[3] *Donald Knuth*, The Art of Computer Programming, Seminumerical Algorithms, Volume 2, 3rd edition, Addison Wesley, Reading, Massachusetts, 1998.
[4] *P. L'Ecuyer and R. Simard*, TestU01: A C library for empirical testing of random number generators, ACM Transactions on Mathematical Software, 33, 4, Article 22, 2007.
[5] *George Marsaglia*, DIEHARD Statistical Tests: http://stat.fsu.edu/ geo/diehard.html.
[6] *S. Murphy*, The power of NIST's statistical testing of AES candidates, Preprint. January 17, 2000.
[7] *A. Oprina, A. Popescu, E. Simion, and Gh. Simion*, Walsh-Hadamard Randomness Test and New Methods of Test Results Integration, Bulletin of Transilvania University of Braşov, vol. 2(51) Series III-2009, pg. 93-106.
[8] *** NIST Special Publication 800-22, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications, 2001.
[9] *** NIST standards: http://www.nist.gov/, http://www.csrc.nist.gov/, Randomness Testing of the Advanced Encryption Standard Candidate Algorithms, NIST IR 6390, September 1999.
[10] *C. Georgescu, E. Simion*, New results concerning the power of NIST randomness tests, Proceedings of the Romanian Academy Series A, Vol. 18, 2017.
[11] *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.

[12] *A. Doğnaksoy, 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.
[13] *J. L. Rodgers, W. A. Nicewander*, Thirteen Ways to Look at the Correlation Coefficient, The American Statistician, Vol. 42, No. 1, Feb., 1988.
[14] 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.