Testing Randomness in Ciphertext of Block-Ciphers Using DieHard Tests

Dr.Mohammed M. Alani
Asst. Professor in Computer Engineering Dept.,
College of Computer Engineering and Sciences,
Gulf University, Kingdom of Bahrain.

Summary
One of the important aspects of the security of block ciphers is the randomness of the cipher text. One criterion used to evaluate the Advanced Encryption Standard (AES) candidate algorithms was their demonstrated suitability as random number generators. In this paper, we introduced a new approach to interpret the results of tests of randomness. This approach relies on using DieHard battery of tests which was designed originally to test the randomness of Random Number Generators (RNGs). The proposed approach was used to test the randomness of five types of data; plaintext, 3DES-encrypted, AES-encrypted, Serpent-encrypted, and Blowfish-encrypted. The tests resulted in 47 p-values for each type of data. This paper suggests the classification of results of DieHard tests into three areas; Safe Area, Doubt Area, and Failure Area. The resulting p-values for each data type were distributed over these areas according to the suggested ranges. The data type having more p-values in the Safe Area indicate better randomness, while the existence of many p-values in the Failure Area indicates deviation from randomness. The results of the implemented tests showed that AES- and Blowfish-encrypted data provided equal results in term of number of p-values distributed over different areas. The AES and Blowfish results were slightly better than Serpent-encrypted data while 3DES encrypted data had more p-values in the Doubt Area.

Keywords:
randomness, diehard tests, aes, blowfish, block cipher

1. Introduction

Block ciphers have evolved rapidly in the last two decades. For a long time Data Encryption Standard (DES) was the dominating block cipher, until it started showing signs of aging with the rapid increase in the capabilities of modern computers [1]. Along with the huge development of computers new threats evolve. The same machines that provide security that was not available before can perform attacks and threats to this security.

One of the important aspects of the security of block ciphers is the randomness of the cipher text. One of the criteria used to evaluate the Advanced Encryption Standard (AES) candidate algorithms was their demonstrated suitability as random number generators. That is, the evaluation of their output utilizing statistical tests should not provide any means by which to computationally distinguish them from a truly random source.

Previous analysis of randomness of block ciphers focused mainly on avalanche tests [2]. These tests were based on NIST statistical tests [3]. Another paper discussed the use of DieHard battery of tests on AES, but the results were not given a deep explanation or formalized in a clear way [4].

In this paper, we apply DieHard Battery of Tests to the ciphertext produced by various block ciphers [5]. This group of tests consists of 12 tests that are usually used to test randomness of Random Number Generators (RNG). Furthermore, the results are categorized in a way to give clear indicators of randomness of cipher-text produced by the tested block ciphers.

The DieHard tests employ chi-squared goodness-to-fit technique to calculate a p-value. This p-value should be uniform on [0,1) if the input file contains truly independent random bits. Those p-values are obtained by p=F(X), where F is the assumed distribution of the sample random variable X; often normal. But that assumed F is just an asymptotic approximation, for which the fit will be worst in the tails. This means that p-values near 0 or 1 indicate deviation from normal distribution. In other words, chi squared goodness-to-fit is applied to indicate how close the distribution of the tests' results to normal. On the other hand, the tests themselves try to give indicators to the randomness of the bit-stream that is considered the test space.

This paper focuses on providing simple and clear interpretation of the DieHard tests' results and gives measurable indicators of randomness of the ciphertext.

2. DieHard Randomness Tests

The DieHard tests battery includes twelve tests of which few are repeated with different parameters. These tests are:

Birthday spacing: Choose random points on a large interval. The spacings between the points should be asymptotically Poisson distributed. The name is based on the birthday paradox [6].

Overlapping permutations: Analyze sequences of five consecutive random numbers. The 120 possible orderings should occur with statistically equal probability.

Ranks of matrices: Select some number of bits from some number of random numbers to form a matrix over {0,1}, then determine the rank of the matrix. The count of the ranks should follow a certain distribution.

Manuscript received April 5, 2010
Manuscript revised April 20, 2010

1. Monkey tests (bit stream tests): Treat sequences of some number of bits as "words". Count the overlapping words in a stream. The number of "words" that don't appear should follow a known distribution. The name is based on the infinite monkey theorem [7].
2. Count the 1s: Count the 1 bits in each of either successive or chosen bytes. Convert the counts to "letters", and count the occurrences of five-letter "words".
3. Parking lot test: Randomly place unit circles in a 100 x 100 square. If the circle overlaps an existing one, try again. After 12,000 tries, the number of successfully "parked" circles should follow a certain normal distribution.
4. Minimum distance test: Randomly place 8,000 points in a 10,000 x 10,000 square, and then find the minimum distance between the pairs. The square of this distance should be exponentially distributed with a certain mean.
5. Random spheres test: Randomly choose 4,000 points in a cube of edge 1,000. Center a sphere on each point, whose radius is the minimum distance to another point. The smallest sphere's volume should be exponentially distributed with a certain mean.
6. The squeeze test: Multiply 231 by float random integers on [0,1) until you reach 1. Repeat this 100,000 times. The number of floats needed to reach 1 should follow a certain distribution.
7. Overlapping sums test: Generate a long sequence of random floats on [0,1). Add sequences of 100 consecutive floats. The sums should be normally distributed with characteristic mean and sigma.
8. Runs test: Generate a long sequence of random floats on [0,1). Count ascending and descending runs. The counts should follow a certain distribution [8].
9. The craps test: Play 200,000 games of craps, counting the wins and the number of throws per game. Each count should follow a certain distribution.
10.

Since we are trying to test for randomness, all the counts produced from these test are expected to have close-to-normal distribution for better randomness. For this purpose, the chi squared goodness-to-fit is used to produce a p-value that should be uniform on [0,1). Those p-values are obtained by p=F(X), where F is the assumed distribution of the sample random variable X; often normal. But that assumed F is just an asymptotic approximation, for which the fit will be worst in the tails. This implies that a p-value near 0 or 1 is an indicator of deviation from randomness.
Some of the tests performed yielded more than one p-value. In some of these cases a Kolmogorov–Smirnov (KS) test was run one the p-values to produce a single p-value that indicates randomness [9].
Count the 1s test for specific bytes was performed but its results were not included in the results section of this paper because the number of p-values produced by this tests was large and these tests had to be repeated for several encryption algorithms which may lead to excessively long results section. Thus, the results of this test were not included in this paper. On the other hand, the results of two of the tests, birthday spacing and monkey tests, were detailed because they are considered two of the most important test in the DieHard tests battery [10].

3. Block Ciphers and Test Data

The following four block ciphers were chosen to undergo the randomness test in this research:

1. 3DES (Triple-DES) with 192-bit key
2. Advanced Encryption Standard (AES) with 256-bit key.
3. Serpent with 256-bit key.
4. Blowfish with 448-bit key.

Both AES and Serpent were previously tested for randomness during the AES selection phase [2].
Since DieHard tests require a set of data with the size of 10 and 12 MBytes, the test data was chosen to have the size of 13.718Mbytes. This test data came from a combination of many types of file such as executable programs, audio, video, and text. For randomness tests, the sample space of more than 100,000 bits is considered reasonable.
The test data was encrypted with the four types of encryption algorithms mentioned above into four separate files.

4. Tests Results

The plaintext file as well as the four encrypted files were subjected to the DieHard tests and gave the results shown in tables 1, 2, 3, 4, and 5 below.

Table 1: DieHard Tests Results for the Plaintext file

Test Name	p-value	Test Name	p-value
Birthday pacing	1.000000	Monkey Test	1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
Overlapping	1.000000		1.000000

Permutations	1.000000		1.000000
Binary Rank	1.000000		1.000000
	1.000000		1.000000
	1.000000		1.000000
Count the 1's	1.000000		1.000000
	1.000000		1.000000
3D Spheres	1.000000		1.000000
Squeeze	1.000000		1.000000
Overlapping Sums	1.000000		1.000000
Craps	1.000000		1.000000
	0.999997	Runs	1.000000
Min. Distance	1.000000		1.000000
Parking Lot	1.000000		1.000000
			1.000000

Table 2: DieHard Tests Results for the 3DES-encrypted file

Test Name	p-value	Test Name	p-value
Birthday Spacing	0.742890	Monkey Test	0.91658
	0.893112		0.07795
	0.559102		0.78356
	0.850000		0.16944
	0.104397		0.43842
	0.620756		0.93809
	0.267198		0.03979
	0.748315		0.24436
	0.856381		0.76818
Overlapping Permutations	0.780205		0.94594
	0.145390		0.75295
Binary Rank	0.389660		0.85963
	0.780205		0.25775
	0.159133		0.57658
Count the 1's	0.599488		0.65755
	0.305525		0.16651
3D Spheres	0.928702		0.75736
Squeeze	0.542369		0.86222
Overlapping Sums	0.834684		0.28326
Craps	0.523433		0.18585
	0.023614	Runs	0.310885
Min. Distance	0.541504		0.752972
Parking Lot	0.238872		0.819962
			0.298053

Table 3: DieHard Tests Results for the AES-encrypted file

Test Name	p-value	Test Name	p-value
Birthday Spacing	0.764805	Monkey Test	0.64197
	0.310685		0.76317
	0.333483		0.18398
	0.687379		0.96262
	0.036211		0.31987
	0.420004		0.88587
	0.168696		0.03091
	0.767771		0.81122
	0.267583		0.46618
Overlapping Permutations	0.530497		0.28484
	0.999037		0.14852
Binary Rank	0.326572		0.41916
	0.385797		0.70826
	0.318936		0.55267
Count the 1's	0.983250		0.99172
	0.301962		0.67958
3D Spheres	0.441769		0.81185
Squeeze	0.876453		0.25926
Overlapping Sums	0.012551		0.05466
Craps	0.787202		0.68707
	0.359208	Runs	0.377076
Min. Distance	0.173347		0.613504
Parking Lot	0.861114		0.587110
			0.037080

Table 4: DieHard Tests Results for the Serpent-encrypted file

Test Name	p-value	Test Name	p-value
Birthday Spacing	0.685417	Monkey Test	0.69529
	0.626883		0.74999
	0.927469		0.48385
	0.111697		0.94517
	0.116412		0.54897
	0.918441		0.38394
	0.410264		0.24362
	0.002424		0.83135
	0.839370		0.79370
Overlapping Permutations	0.757021		0.02387
	0.474396		0.48758
Binary Rank	0.631214		0.36710
	0.321199		0.80160
	0.962556		0.84059
Count the 1's	0.268669		0.23636
	0.477565		0.47361
3D Spheres	0.685843		0.59389
Squeeze	0.219412		0.64022
Overlapping Sums	0.322358		0.80095
Craps	0.671267		0.06031
	0.147195	Runs	0.532185
Min. Distance	0.935313		0.999528
Parking Lot	0.235249		0.318428
			0.389448

Table 5: DieHard Tests Results for the Blowfish-encrypted file

Test Name	p-value	Test Name	p-value
Birthday Spacing	0.527912	Monkey Test	0.77734
	0.515234		0.09211
	0.738829		0.23564
	0.520157		0.64458
	0.398417		0.14797
	0.217241		0.08427
	0.827327		0.92938
	0.202970		0.64197
	0.249973		0.17783
Overlapping Permutations	0.073719		0.41369
	0.712404		0.01802
Binary Rank	0.469144		0.44857
	0.351896		0.12120
	0.463594		0.64805
Count the 1's	0.852200		0.07863
	0.936581		0.87419

3D Spheres	0.590078		0.84509
Squeeze	0.542446		0.59207
Overlapping Sums	0.685277		0.28010
Craps	0.638317		0.34705
	0.195471	Runs	0.730297
Min. Distance	0.352774		0.266386
Parking Lot	0.972924		0.980337
			0.463899

5. Interpretations of Results

In order to have a clearer view of the results, we suggest defining population areas in the [0,1) range where the p-values are distributed. We suggest dividing this range into three types of areas; Safe Area, Doubt Area, and Failure Area. These areas can be defined by the following limits:
0<p-value≤0.1 or 0.9≤p-value≤1 fall in the Failure Area.
0.1<p-value≤0.25 or 0.75≤p-value<0.9 fall in the Doubt Area.
0.25<p-value<0.75 fall in the Safe Area.
These areas can be expressed in a graphical way as in Figure 1.

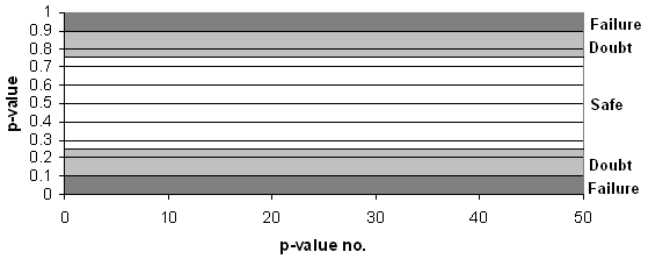

Fig. 1 Safe, Doubt, and Failure Areas

Having more p-values in the Safe Area indicates that the tested sample is closer to randomness. On the other hand, having to many p-values in the Failure Area is an indicator of deviating from randomness.
The various tests included in the DieHard battery vary in their importance and hardness as randomness tests [10]. Thus, the number of p-values we included in the results tables for each test can be thought of as the weight of this test as a part of a set of 47 p-values that we calculated for each encryption algorithm. Hence, the Monkey test is considered the most important by having 20 p-values out of the total 47 p-values. The birthday spacing tests are considered the second in importance having 9 p-values out of the total 47, and so on for the other tests.
Table 6 shows the number of p-values in each area for each encryption algorithm. Figures 2,3,4, and 5 show the p-values for 3DES, AES, Serpent, and Blowfish consecutively. The p-values of the plain text were not put into a figure because all of its p-values fall in the Failure Area.

Table 6:Number of p-values in each area

Data type	Number of p-values in the Safe Area	Number of p-values in the Doubt Area	Number of p-values in the Failure Area
Plaintext	0	0	47
3DES encrypted	18	22	7
AES encrypted	25	13	9
Serpent encrypted	24	14	9
Blowfish encrypted	25	13	9

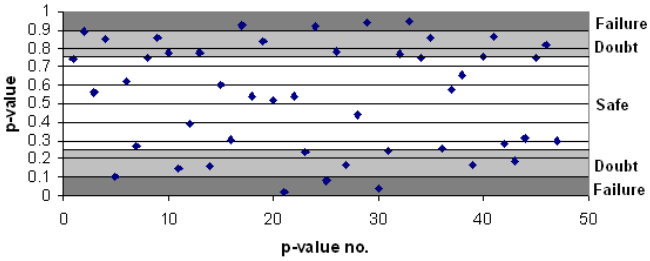

Fig. 2. p-values for 3DES-encrypted data

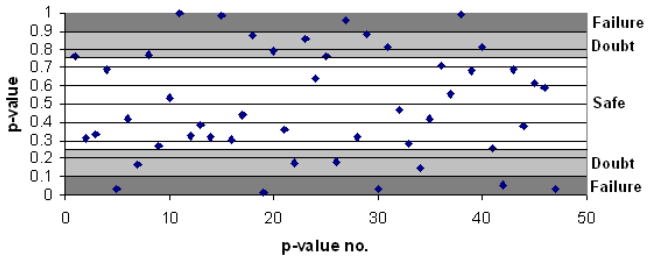

Fig. 3. p-values for AES-encrypted data

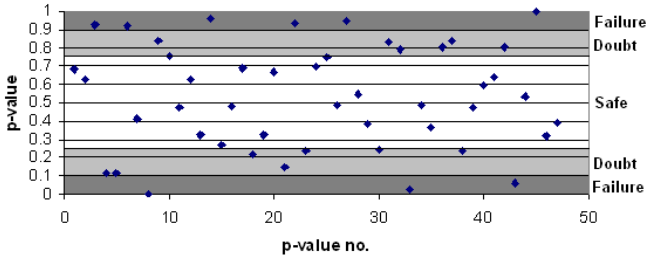

Fig. 4. p-values for Serpent-encrypted data

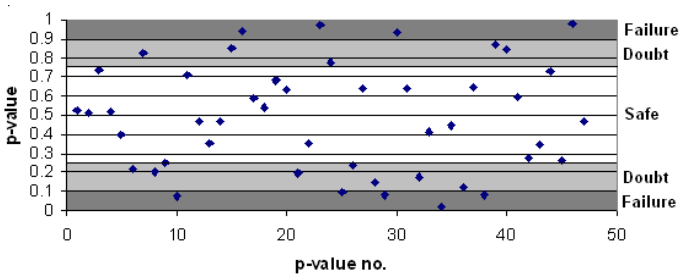

Fig. 5. p-values for Blowfish-encrypted data

6. Conclusions

In this paper, we introduced a new approach to interpret the DieHard tests applied to test the randomness of block ciphers. This approach relies on using DieHard battery of tests which was designed originally to test the randomness of RNGs. This approach was used to test the randomness of five types of data; plaintext, 3DES-encrypted, AES-encrypted, Serpent-encrypted, and Blowfish-encrypted. These test resulted in 47 p-values for each type of data. These p-values were in the range [0,1). p-values close to 0 or 1 indicate deviation from randomness.

This paper focuses on introducing the results in a clear and simplified way. A new classification of p-values was suggested. The p-values range was divided into three areas; Safe Area, Doubt Area, and Failure Area. The resulting p-values for each data type were distributed over these areas according to the suggested ranges. The data type having more p-values in the safe area indicate better randomness, while the existence of many p-values in the Failure area indicates deviation from randomness. This interpretation of results provides a way to conclude measurable indicators to randomness of block ciphers.

The results of the implemented tests showed that AES- and Blowfish-encrypted data provided equal results in term of number of p-values distributed over different areas. The AES and Blowfish results were slightly better than Serpent-encrypted data while 3DES encrypted data had more p-values in the Doubt Area.

References

[1] Walter Tuchman, "A brief history of the data encryption standard". Internet besieged: countering cyberspace scofflaws. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA. pp. 275–280 (1997).

[2] J. Soto, Jr.,Randomness Testing of the Advanced Encryption Standard Candidate Algorithms, NIST.

[3] NIST Special Publication 800-22 Revision 1 (Aug. 2008). http://csrc.nist.gov/publications/nistpubs/800-22-rev1/SP800-22rev1.pdf

[4] The Marsaglia Random Number CDROM, with The Diehard Battery of Tests of Randomness, produced at Florida State University under a grant from The National Science Foundation (1985).

[5] Peter Hellekalek , Stefan Wegenkittl, Empirical evidence concerning AES, ACM Transactions on Modeling and Computer Simulation (TOMACS), v.13 n.4, p.322-333, October 2003.

[6] E. H. McKinney, "Generalized Birthday Problem", American Mathematical Monthly No 73, pp: 385–387 (1996).

[7] Isaac, Richard E., The Pleasures of Probability. Springer. pp. 48–50. ISBN 038794415X (1995).

[8] A. Wald, and J. Wolfowitz, SAS Institute, SAS Samples & NotesSample 33092: Wald-Wolfowitz (or Runs) test for randomness.

[9] T.T.Soong, Fundamentals of Probability and Statistics for Engineers, John-Wiley and Sons Ltd., pp. 327., ISBN: 0470868147 (2004).

[10] G. Marsaglia, W. W. Tsang, "Some difficult-to-pass tests of randomness", Journal of Statistical Software, Vol. 7, Issue 3, (Jan. 20020.

Dr.Mohammed M. Alani received his BSc., MSc., and PhD. from Al-Nahrain University, Baghdad, Iraq. He worked as an IT consultant for Middle-East Communications in Amman, Jordan and as a Cisco Academy Instructor as well 2005-2007. He worked as an assistant prof. in ComputerMan college in Sudan. Currently he is working as an assistant prof. in Gulf University in Bahrain. He has many industrial certificates from many well-known information technology companies such as Cisco, Microsoft, and Brocade.