Grzegorz GANCARCZYK
Agnieszka DĄBROWSKA-BORUCH
Kazimierz WIATR

STATISTICS IN CYPHERTEXT DETECTION*)

ABSTRACT *Mostly when word encrypted occurs in an article text, another word decryption comes along. However not always knowledge about the plaintext is the most significant one. An example could be a network data analysis where only information, that cipher data were sent from one user to another or what was the amount of all cipher data in the observed path, is needed. Also before data may be even tried being decrypted, they must be somehow distinguished from non-encrypted messages.*

In this paper it will be shown, that using only simple Digital Data Processing, encrypted information can be detected with high probability. That knowledge can be very helpful in preventing cyberattacks, ensuring safety and detecting security breaches in local networks, or even fighting against software piracy in the Internet.

Similar solutions are successfully used in steganalysis and network anomaly detections.

Key words: *cipher, cryptography, cryptanalysis, ciphertext, encryption, data analyzer, data distribution, white noise, statistics.*

*) The work presented in this paper was financed by The National Centre for Research and Development through the research program – 'SYNAT' SP/I/1/77065/10.

Grzegorz GANCARCZYK, M.Sc.
e-mail: g.gancarczyk@cyfronet.pl

Agnieszka DĄBROWSKA-BORUCH, Ph.D.
e-mail: adabrow@agh.edu.pl

Prof. Kazimierz WIATR
e-mail: wiatr@agh.edu.pl

AGH University of Science and Technology; Faculty of Electrical Engineering, Automatics, Computer Science and Electronics; Department of Electronics

ACC CYFRONET AGH

PROCEEDINGS OF ELECTROTECHNICAL INSTITUTE, Issue 251, 2011

1. PRELIMINARIES

Cryptography is one of the most significant part of modern Computer Science. It is located on the border of Mathematics, Computer Science and Electronics. It evolves within evolution of those sciences.

Eighty years ago it was a domain of mathematicians only. With advance in technology and birth of new disciplines, like mentioned Computer Science or Telecommunication, it moved from one science discipline to another. Nowadays, it is seen as part of the Computer Science.

It is told, that Great War was a war of Chemistry, Second World War was a war of Physics and the next great war will be a war of Cryptography.

It is not a fool assumption, if we look in the past and see, that till the Nineteen Nineties in the U.S. Cryptography was seen as a weapon and limited in use by the government.

Cryptography and Cryptanalysis can be used to serve *right* or *wrong*.

Encrypted messages were used by terrorists and criminals to communicate each other (e.g. March 20th 1995 – attack in Tokyo's subway, 12 people killed, over 6.000 injured) [1]. Illegal software and intellectual property are being downloaded as ciphertext using peer-to-peer network clients to prevent detection of such an act of abuse [2]. Child Pornographers and Paedophiles use both encrypted and hidden (using *steganography* techniques) files/information (e.g. June 1997, hearings of U.S. senator Charles Grassley about sexual molestation of children) [1, 3].

To prevent those and other actions, sophisticated methods of detection were and are in use. A group of two is significant for this paper to show current state-of-the-art.

First kind of methods is based on detection of *abnormal behaviour* of local networks. Registering real time network bandwidth usage and total amount of downloaded by single user data is recommended in [2], as it can be useful in preventing crime.

S. Mika proposes in [4] to make a model of behaviour of 'healthy' network and then to monitor it. While monitoring, not only number and type of sent packets should be checked, but also their headers and payloads. Just like in [5].

Second kind of methods is based on detection of *additional* and/or *special type* of data. They use simple statistic parameter like *entropy* or *data distribution* (histogram shape) [6].

The simpler the better method is, because it can be implemented not only as a software, but also as a System on Chip solution [7].

Main motivation for this work was to find such an algorithm of encrypted information detection, which would be:

- possible in implementation both in hardware and software solutions,
- based on pure and easy logical and arithmetical operations,
- based on digital data,
- independent from analysed data type (i.e. *text*, *video*, etc. files),
- immune for data partitioning (e.g. during network transmission) and data routing (i.e. accurate to say about whole information encryption, only by analysing one part of it).

Such detector of encryption may be used to monitor anomalies in observed path/channel and point the source and the destination of information being cause of that abnormality. There are many kinds of computer networks, where *all* or *almost all* traffic should (e.g. banking, national security, medical service, military) or should not be (e.g. home, public, overall access) encrypted. It can be also used to reveal, which intercepted data from *stream* or *storage medium* are plaintext, and which are ciphertext, that should be decrypted.

2. INTRODUCTION

If encrypted data should be properly detected, then firstly their characteristic properties must be well defined. Let us take a look at modern methods of securing information against unauthorised access.

In most cases plaintext is being encrypted by one of many commonly used digital data ciphers. The type of used key (*symmetric, asymmetric*) is not important here. Much more interesting is, how data are being encrypted (*stream, block* or *block* & *stream* encryption algorithm) and the topology of the cipher. There are three commonly used topologies:

- Feistel networks (e.g. 3DES) [8, 9],
- Substitution–Permutation networks (*S–P networks*, e.g. AES) [8],
- logic operations stream networks (e.g. RC4) [8, 10].

The role of *black boxes* (*permutation boxes, substitution boxes, expansion boxes*) in cipher core is to obtain *confusion and diffusion* firstly mentioned by Claude E. Shannon [9, 11]. In his work from 1949 – *Communication Theory of Secrecy System* – Shannon not only writes about, how to obtain confusion and diffusion, but also defines *The Perfect Secrecy*.

It is assumed by authors, that output data of cipher (ciphertext) should have similar to *Shannon's Perfect Cipher* values of statistic parameters.

3. PERFECT SECRECY

Condition to obtain ideal secure ciphertext is by usage of the *one–time pad* key [8,11–14]. It means, that the key used in encryption process must be fully random (must have fully random values) and have length at least equal to plaintext length. If this condition is fulfilled, then for cipher output message following feature occurs – *no output data value can be more possible then others* [11]. In other words, ciphertext has uniform distribution.

In real world cipher algorithm core operate using only pseudorandom number generator with finite output (*key*) length (typically 128, 256 or 512 bits). The condition of confusion and diffusion is fulfilled better or worse.

That is why it will be spoken only about *nearly Perfect Secrecy*. Nevertheless, assumption that real ciphers should product as uniformly distributed random output data, as it is possible, was made.

4. DATA

There are many ways of interpreting data. In our work an assumption was made, that the smallest portion of information is a byte/octet (8 bits). It is made that way, because in network data transmission or data storage byte/octet is the smallest used unit [15]. Other reason is, that in ASCI code 8 bits are used to code a single sign.

Byte of data is treated as unsigned integer with values in range from 0 to 255.

Analysed data can be always divided in other ways (e.g. on 32 bits unit), but no other division is as optimal, as 8 bits unit is. In case of division different than 1 bit, 2, 4 or 8 bits a non-integer number of divisions always occur. Smaller the unit size is, more operations must be done and more time for data processing is wasted.

5. STATISTIC PARAMETERS

There are many statistic parameters, but not all of them are useful in random data processing. A group of fourteen useful was formed. They are commonly used in digital signal processing [16], computer methods of objects

identification [17] and statistic analysis of data [17]. Values of those are used as references in proposed *Encrypted Data Detection Algorithms*. Those are:

- Modified Mean Value,
- Energy,
- Mean Power,
- Root Mean Square Value,
- Variation,
- Standard Deviation,
- Modified Variation,
- Modified Standard Deviation,
- Variations Difference,
- Standard Deviations Difference,
- Normal Moments from 0^{th} to 5^{th},
- Central Moments from 0^{th} to 5^{th},
- Mean Spectral Power,
- Modified Histogram.

Names of algorithms used in any proposed *Encrypted Data Detector*, will be written in short as names of used in them statistic parameters and using upper cases and *italic type*. Names of statistic parameters will be written using only lower cases.

Mathematical formulas of all *non-modified* parameters can be found in [16]. Definitions of all *modified* parameters will be presented in chapter 6 *Definitions*.

Not all from the parameters listed above can be named *statistic* (e.g. energy), but most of them can. That is why the whole group will be called from now on *Statistic Parameters*.

Adjective *modified* means, that the definition of a parameter used in detection process slightly differs from its original form.

6. DEFINITIONS

Definitions of all unclear *Statistic Parameters* are presented here.

Mean value of discrete function is given by expression

$$\bar{x} = \frac{1}{N}\sum_{n=0}^{N-1} x(n) \ , \tag{1}$$

where N is the number of all samples, $x(n)$ is value of sample with index n.

Used in the detection process parameter is called *Modified Mean Value* and it is calculated in two steps.

The first one is to calculate the mean value for all given data. If error (calculated for reference level equal to 127.5) is positive, then data are classified as encrypted. If error is negative and greater than −2%, then *mean value in window* is being calculated. In other cases data are classified as not encrypted.

In window means, that mean value is being calculated in predefined window. The definition of *mean value in window* is

$$\bar{x} = \frac{1}{W}\sum_{n=0}^{W-1} x(n) \ , \tag{2}$$

where W is window size. Window size is always multiple of 2. Minimal window size is 2, maximum cannot be greater than data length N.

When value of (2) was calculated, window moves by 1 (series (2) limits in next step are n = 1 and n = W), next mean value in window is calculated and so on. When window reaches data end (series (2) limits are n = N−W−1 and n = N−1) calculations are over. From all gathered mean value in window results a mean value is calculated. Process is repeated for all possible window W values.

If error sign changes at least once for any window W length, then analysed data are classified as encrypted.

Variation of data is given by

$$\sigma_x^2 = \frac{1}{N}\sum_{n=0}^{N-1}\left[x(n) - \bar{x}\right] . \tag{3}$$

Modification of this parameter is a result of two ways of interpreting the mean value in (3). First one (non-modified) uses mean value calculated using all available data in considered case [using equation (1)]. Second one (modified) uses theoretical mean value, which is equal to 127.5. *Modified Variation* definition is

$$\sigma_x^2 = \frac{1}{N}\sum_{n=0}^{N-1}\left[x(n) - 127.5\right] . \tag{4}$$

Variations Difference is equal to difference between (3) and (4).

Standard Deviation definition is given below

$$\sigma_x = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}\left[x(n) - \bar{x}\right]} . \tag{5}$$

Like for earlier example, here are also two ways of defining mean value. First one (non-modified) was written above, second one (modified) is

$$\sigma_x = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}\left[x(n)-127.5\right]} \ . \tag{6}$$

Standard Deviations Difference is equal to difference between (5) and (6).
Used *Normal Moments* definition is

$$\overline{k}_x^m = \sum_{n=1}^{N} n^m x(n) \ , \tag{7}$$

where m is moments order. There were checked and used only first six moments (0^{th} to 5^{th}). There is change in series sum borders from equation (7) towards equations (1) to (6). Previously they were 0 and N−1 respectively, now are 1 and N. This had to be done, because no element (sample, data) can be omitted.

Central Moments definition is given by expression

$$\delta_x^m = \sum_{n=1}^{N}\left(n-\overline{k}_x^1\right)^m x(n) \ . \tag{8}$$

Here also only six first orders of parameter were checked and used.

Modified Histogram is obtained in two steps, by counting the relation between number of values in histograms first and last intervals.
It is done for intervals size equal to 64. If relation is lower than 1.25, then data are classified as encrypted, in other case the same relation is calculated for intervals size equal to 128.

If relation in this second step is between 0.75 and 1.25, then data are classified as encrypted. If relation for both histogram intervals length cases was in range 1.25 to 2.00, then the trend of changes is important. If it is decreasing (lower relation value for 128 histogram intervals length, than for 64 histogram interval length case), then data are classified as encrypted. In all other cases data are classified as plain.

Modified Histogram interval sizes were obtained experimentally. First to last interval ratio is known from theory and equal to 1. For 64 and 128 interval sizes best data distinguish results were observed. The same reason was why tolerance levels are equal to ±25% and +100%.

7. RESULTS AND PROPOSED DETECTION ALGORITHM

Two ways to perform *Encryption Detection* are proposed.

First one uses only one parameter from those listed in chapter 5 *Statistic Parameters*. Value of chosen parameter is being compared with its reference value. Reference values of all *Statistic Parameters* are well known and hold in memory unit (block). They were calculated for random data with uniform distribution. If *Statistic Parameter* value is between tolerance borders, then data are being classified as encrypted. In other case detector marks data as not encrypted.

Second algorithm uses more than one *Statistic Parameter* (e.g. *3rd Central Moment*, *Modified Histogram* and *Mean Value*). Value of each parameter is calculated and compared with corresponding reference value. Proper flags are set up. In the end operation of voting is carried out. If most of the flags are set up (equal to logic '1'), then data are classified as encrypted. In other case data are classified as plain.

It is not hard to draw a conclusion, that the second method of detection should be much more effective (in proper detection of data type) than the first one. The biggest disadvantage of second solution is that it needs much more time and consumes more hardware resources than the first one. It is one of the reasons, why parallel computing is preferred.

In Table 1 efficiency for all single methods is shown. Table 2 consists results for multi method algorithms. One of those is also proposed as the final solution to the title problem. Block scheme for it is presented in figure 10. Its advantages and disadvantages are being spoken about in chapter 8 *Conclusions*.

The efficiency is counted due to the expression

$$\eta_{\%} = \frac{100}{N}\sum_{i=1}^{N}\delta_i \ , \tag{9}$$

where

$$\delta_i = \frac{c_i}{c_i + u_i} \ , \tag{10}$$

and where N – number of components, δ_i – efficiency of correctly distinguished i*-th* file type, c_i – all correctly distinguished examples of i*-th* file type, u_i – all incorrectly distinguished examples of i*-th* file type.

Used file types were plain and encrypted:

- network data (possessed from data fields of *UDP, TCP, HTTP* and *TLS* frames),
- text (*.txt ANSI, .txt UTF-8, .txt Unicode, .txt Big Endian* and *.rtf*),

- music (*.aiff, .mid, .mp3, .wav* and *.wma*),
- graphic (*.bmp, .gif, .ico, .jpg* and *.png*),
- video (*.asf, .avi* and *.mov*),
- archive (*.cab, .jar, .rar, .tar* and *.zip*),
- others (*.exe* and *.html*).

The *.html* files are inside group *others*, because theirs syntax differs a lot from that used in files from group *text*.

Full decapsulation of network frame (in this case an *Ethernet frame*) enabled obtaining data from *data field* of the most internal *TCP/IP Stack* protocol. The size of network data was in range 1 to 2323 bytes (typically from 200 to 1500 bytes). In respect to algorithms destination – *Network Data Analyzer/ /Encrypted Data Detector in TCP/IP Networks* – in most cases analyzed data were smaller than 1500 bytes.

The way of efficiency calculation makes final result invariant to different numbers of samples in each file type group. Any type of file is not preferred too.

Seven types of cipher algorithms were checked:

- 3DES,
- AES,
- CAST–128,
- RC4,
- BlowFish,
- TwoFish,
- Serpent.

The type of used cipher has no influence on formulas (9) and (10).

TABLE 1
Single *Statistic Parameter* algorithms efficiency

Statistic Parameter	**Efficiency [%]**
Modified Mean Value	80.61
Energy	91.51
Mean Power	90.26
Root Mean Square	90.26
Variation	88.67
Standard Deviation	87.27
Modified Variation	89.06
Modified Standard Deviation	89.19
Variations Difference	92.20
Standard Deviation Difference	91.91
Normal Moments 0^{th} to 5^{th}	89.91
Central Moments 0^{th} to 5^{th}	91.59
Modified Histogram	91.17
Mean Spectral Power	89.62

Probably the best *Single Statistic Parameter* algorithm from Table 1 is one based on energy. Argue its:

- high efficiency (91.51%),
- much lower memory and resources usage, than for more effective parameters (e.g. *Variations Difference, Standard Deviations Difference* and *Central Moments*),
- nowadays every DSP (*Digital Signal Processor*) and FPGA (*Field Programmable Gate Array*) is equipped in at least one hardware *MAC unit* (Multiply and Accumulate).

If time of computations, dissipated by device power and its size are not critical, then mentioned *Central Moments* and *Variations Difference* are probably the best solution for title problem due to their highest efficiency.

Examples of values for *Energy*, *3rd Central Moment* and *Standard Deviations Difference* for ciphertext and plaintext are shown in Figures 1 to 9. Reference is marked as *solid line*, *Statistic Parameter* values for ciphertext are *pale squares*, for plaintext they are marked as *dark rhombuses*.

For every *Parameter* there are three cases shown in the figures:

- all tested data,
- data with length from a given interval,
- short data.

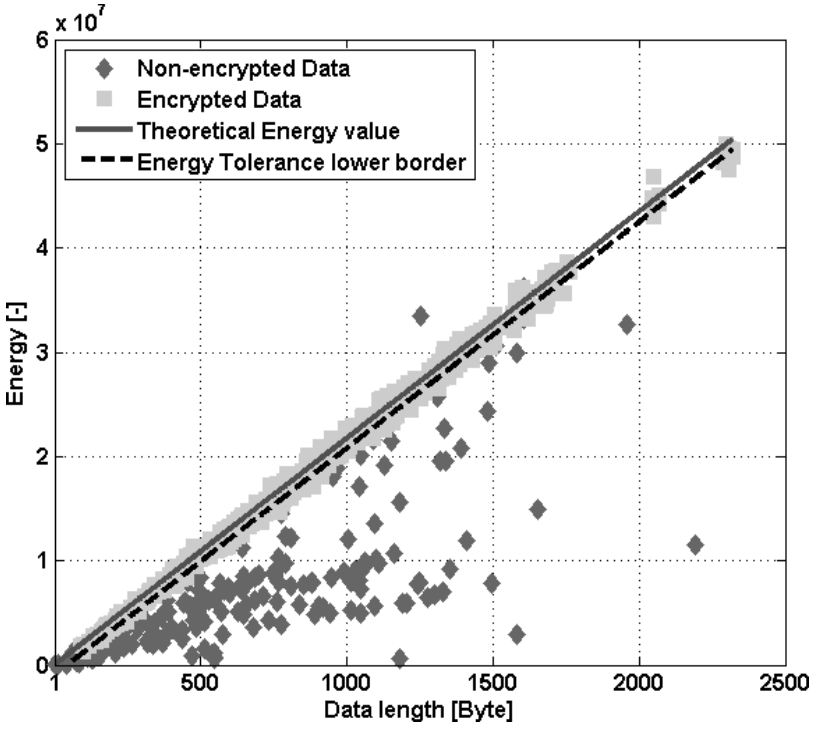

Fig. 1. Encrypted and non-encrypted data *Energy* vs. its length

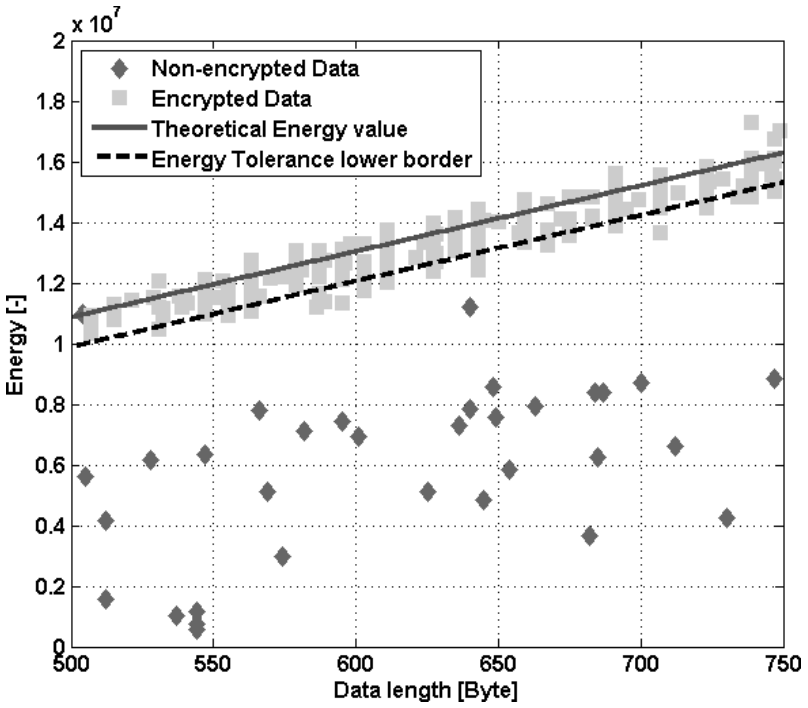

Fig. 2. Encrypted and non-encrypted data *Energy* vs. its length for data from a given interval

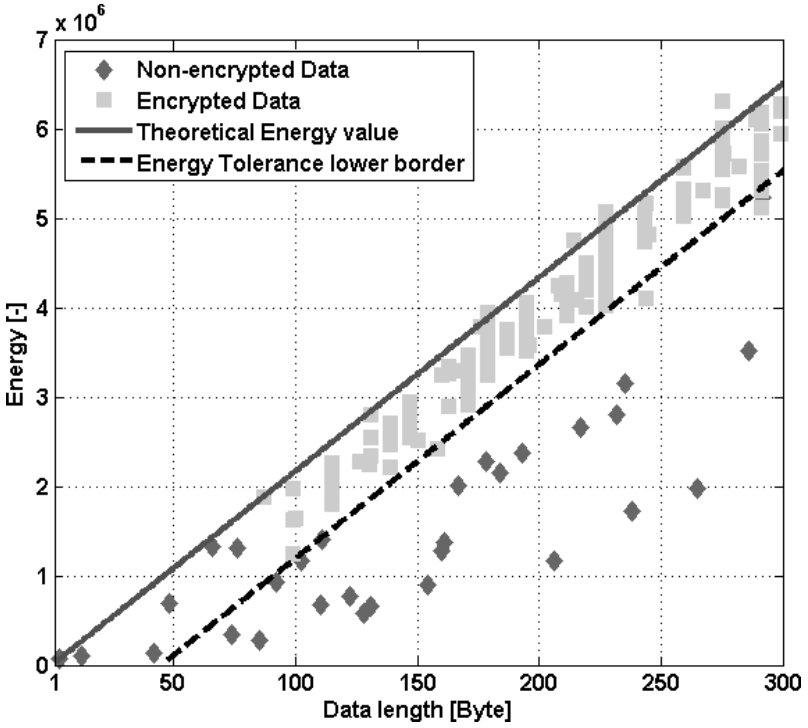

Fig. 3. Encrypted and non-encrypted data *Energy* vs. its length for short data

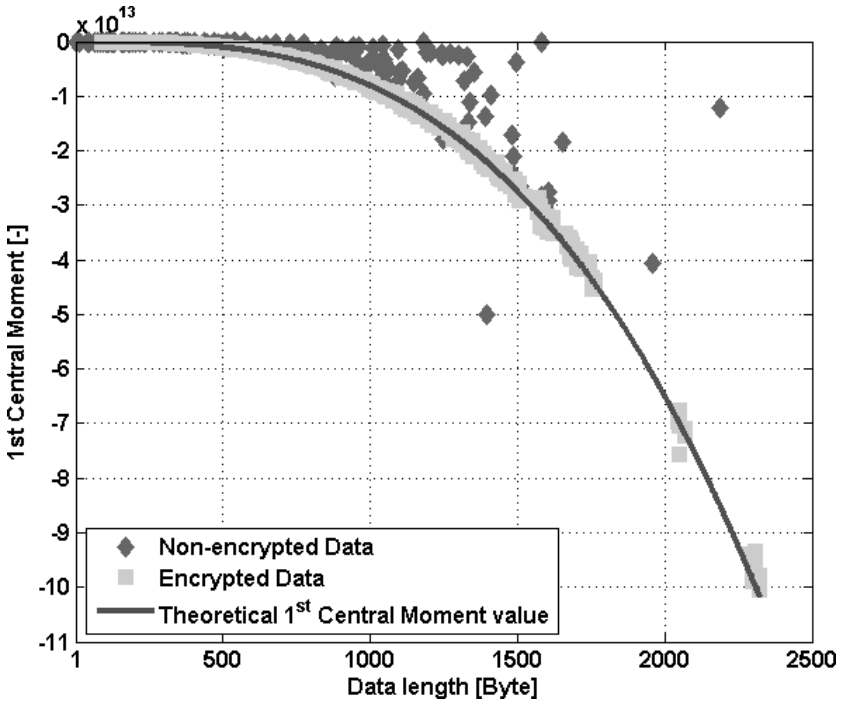

Fig. 4. Encrypted and non-encrypted data *1st Central Moment* vs. its length

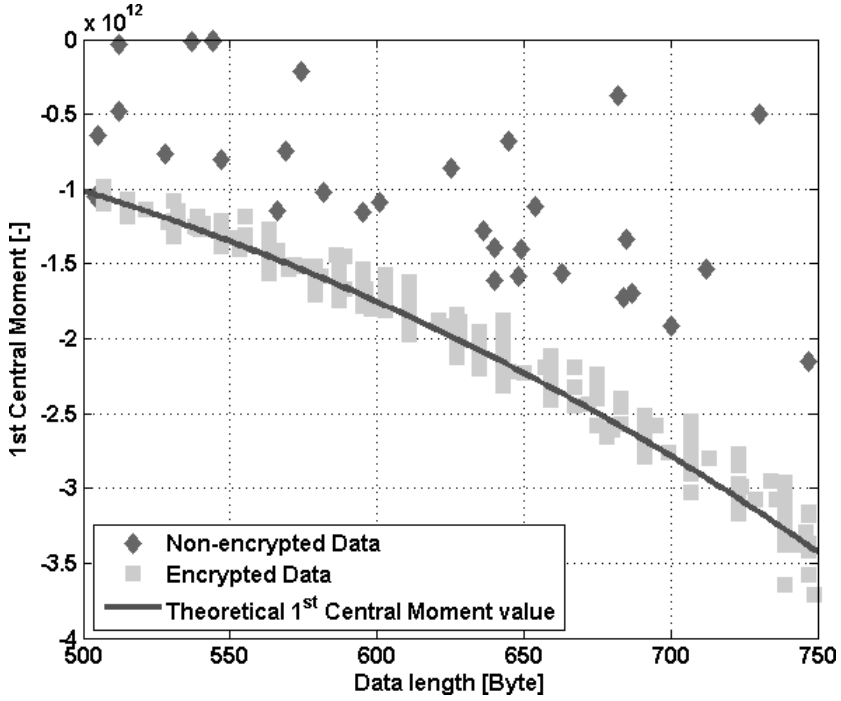

Fig. 5. Encrypted and non-encrypted data *1st Central Moment* vs. its length for data from a given interval

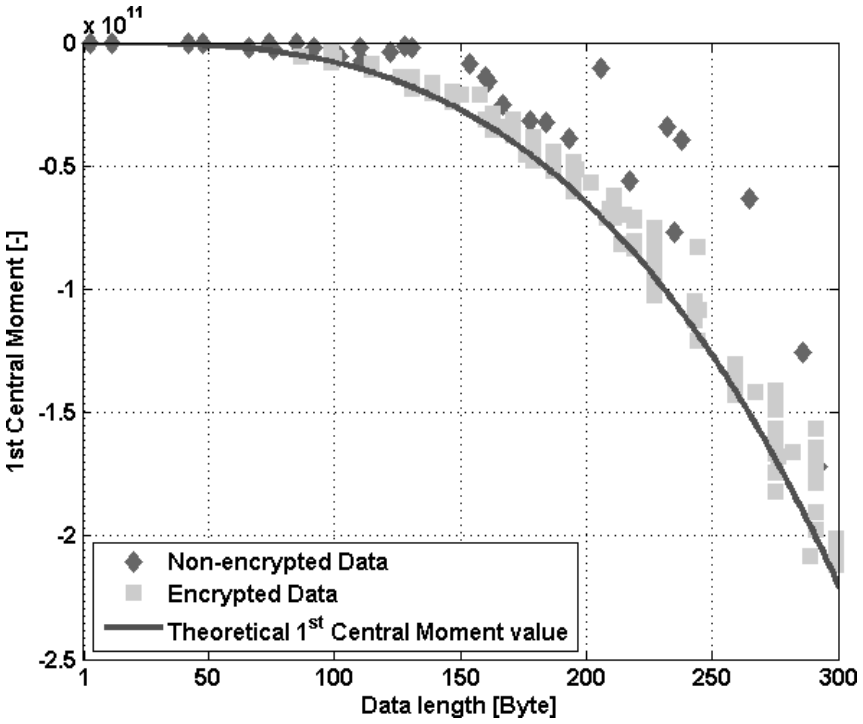

Fig. 6. Encrypted and non-encrypted data *1st Central Moment* vs. its length for short data

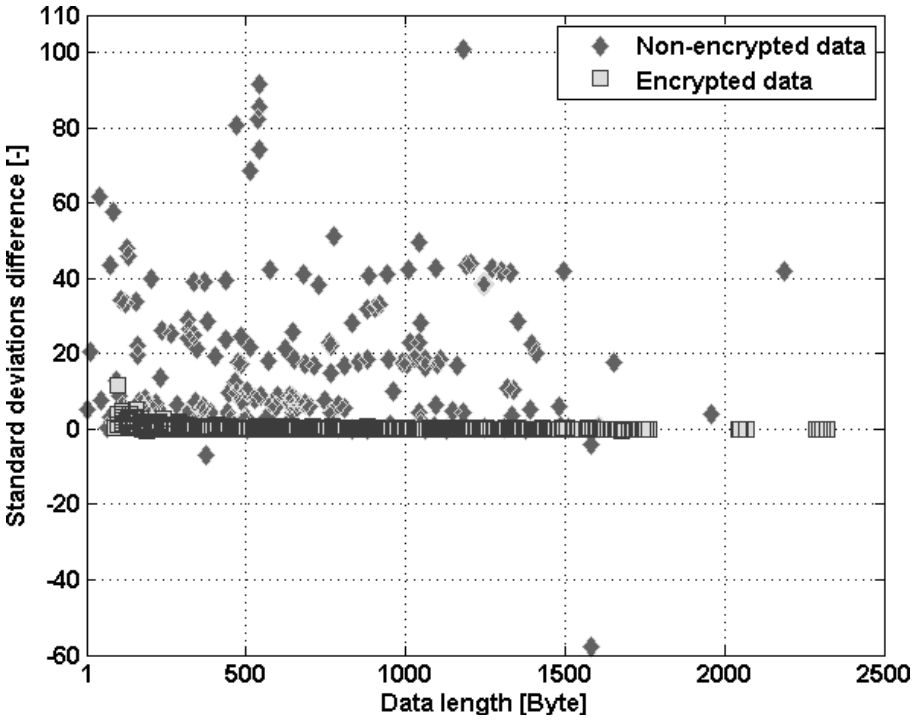

Fig. 7. Encrypted and non-encrypted data *Standard Deviations Difference* vs. its length

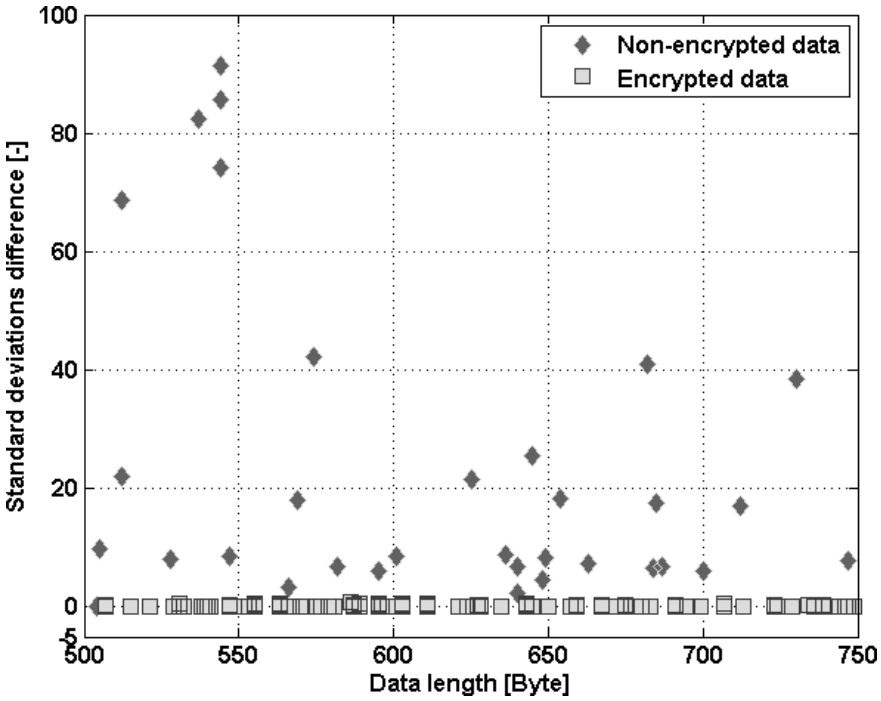

Fig. 8. Encrypted and non-encrypted data *Standard Deviations Difference* vs. its length for data from a given interval

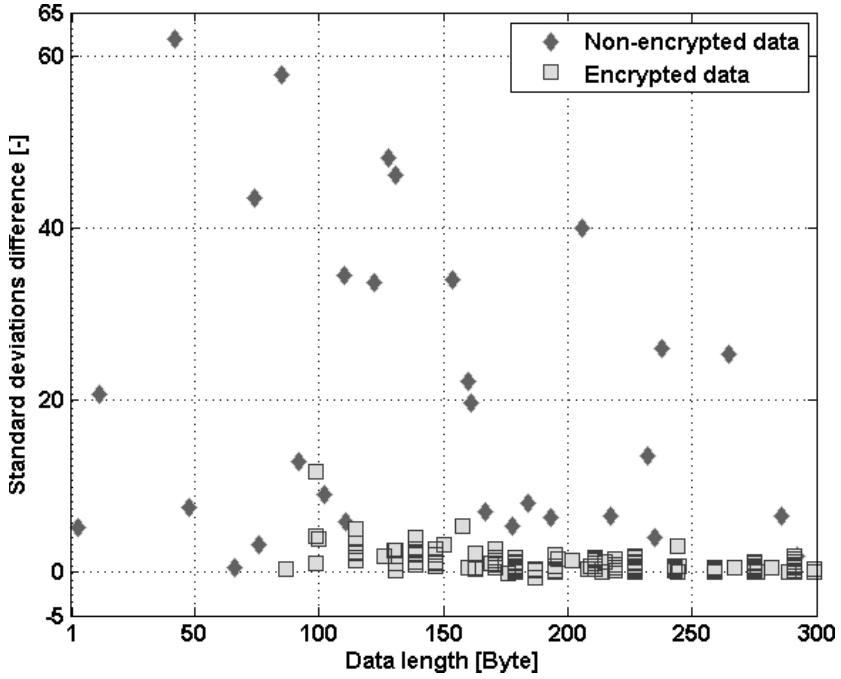

Fig. 9. Encrypted and non-encrypted data *Standard Deviations Difference* vs. its length for short data

Mentioned *Statistic Parameters* values for ciphertext gather near the reference function solid line plot.

Proposed *multi method* algorithm should use *Energy, Standard Deviations Difference* and four first *Central Moments*. Obtained for such solution efficiency is equal to 94.78% and it is greater than for any *single method* algorithm from Table 1.

Efficiency for two proposed *multi method* algorithms is given in Table 2. First one uses most efficient *Statistic Parameters*. Second one, alternative, uses algorithms based on the simplest parameters.

TABLE 2
Multi method algorithm efficiency

Method	Efficiency [%]
Energy & Modified Standard Deviation & Central Moments 0^{th} to 3^{rd}	94.78
Mod. Mean Value & Modified Histogram & Central Moments 0^{th} to 3^{rd}	93.42

Advantage of first *multi method* from Table 2 is its high accuracy. It correctly distinguish 94.78% of used in tests data. It must be noted that efficiency measured by authors for encrypted data only was greater than 99.99%.

As for disadvantage of the method, high time and resources consumption makes it very hard to implement in hardware like FPGA.

Second method efficiency is not as good, as for the previous one, but its simplicity makes up for it. Hardware resources consumption is much more less, therefore smaller (i.e. with lower number of equivalent gates) FPGA chipset can be used to obtain *Encrypted Data Detector*.

For both methods additional calculations of *1^{st} Normal Moment* of the data must be done.

Most time consuming method is that based on *Central Moments*.
Block scheme of proposed *Data Encryption Detector* is given in Figure 10.

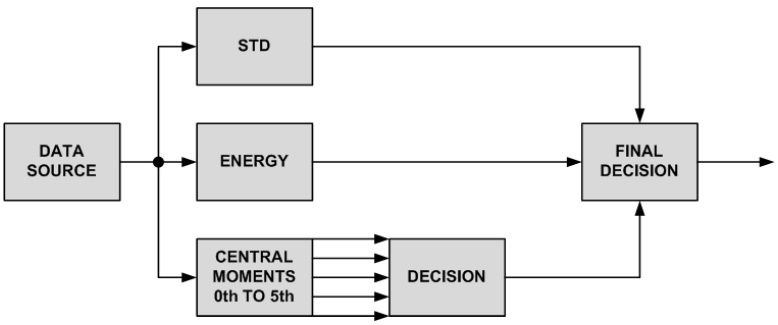

Fig. 10. Block scheme of proposed *Data Encryption Detector*

A *multi method* algorithm, that uses *Energy*, *Modified Standard Deviation*, as well as *Central Moments from 0^{th} to 5^{rd}* was chosen.

Advantages of *Energy* were mentioned earlier.
As for *Central Moments*, they have the best global properties to correctly distinguish data.

Standard Deviations Difference can correctly classifies data, which had been incorrectly classified by one or both earlier methods.

As *DATA SOURCE* binary files or network data can be used. Three next blocks from Figure 10 (*MODIFIED STD, ENERGY, CENTRAL MOMENTS* 0^{th} *TO* 5^{th}) calculate values of parameters corresponding to them. Using *Central Moments from* 0^{th} *to* 5^{th} values, decision about *Central Moment* flag is made. *Final Decision* is made using this flag, *Modified Standard Deviation* and *Energy* flags values.

8. CONCLUSIONS

It was presented, that using Shannon's *Perfect Secrecy* theorem, detection of ciphertext with high probability can be obtained.
Group of so called *Statistic Parameters* was formed.
Two types of detection algorithms were shown. The first one uses only one parameter from formed group. As for the second type, it uses more than one parameter.

There were used three different parameters in presented examples. It can be found in Tables 1 and 2, that usage of only one *Statistic Parameter* makes the detector accuracy poor (in most cases). By poor accuracy, efficiency lower than 90% is meant. When more than one parameter is used in algorithm, then occurred efficiency is better, but also more processing unit resources are being needed.

Like in every statistic process, uncertainty of algorithm detection results is higher for short data.

That is why lower border of methods usage should be created. Proposed level is 80 bytes of data. Below that value number of correctly distinguished plaintext drops below 50% (see Figures 2, 5 and 8).

Methods were not tested for data longer than 2323 bytes. For data with length higher than 1500 bytes they gave good results. That is why there were not any reasons to create higher border of methods usage. In theory more accurate value of any statistic parameter can be calculated, if there are more data.

Proposed methods of cipher detection need only data to produce answer. No other additional information is needed, but it does not mean, that it cannot be used in algorithm modifications to obtain better efficiency (e.g. knowledge about used Transport Layer ports or file extension).

LITERATURE

1. Chang V., Baron D.P.: Sophis Networks and Encryption Export Controls (A), Graduate School of Business, Stanford University, case no. SP–34 (A), 2000.
2. http://gsbapps.stanford.edu/cases/documents/P34A.pdf.
3. Lou X., Fellow K.: Collusive Piracy Prevention in P2P Content Delivery Networks. IEEE Transactions on Computers, vol. 58, no. 7, pp. 970-983, 2009.
4. Astrowsky G.H.: Steganography: Hidden Images, A new Challenge in the Fight Against Child Porn, 2011.
5. http://www.antichildporn.org/steganog.html.
6. Mika S.: Koncepcja hybrydowego systemu detekcji robakow sieciowych wykorzystującego metody eksploracji danych. Metody Informatyki Stosowanej, vol. 23, no. 2/2010, pp. 105--115, 2010. (not available in English).
7. Cheema F.M., Akram A., Iqbal Z.: Comparative Evaluation of Header vs. Payload based Network Anomaly Detectors. Proceedings of the World congress on Engineering, vol. 1 WCE 2009, 2009.
8. Składkiewicz M.: Entropia – pomiar i zastosowanie. Hakin9, no. 3, 2008. (not available in English).
9. http://www.hakin9.org.
10. Damiani E., Dipanda A., Yetongnon K., Legrand L., Schelkens P., Chbeir R.: Signal Processing for Image Enhancement and Multimedia Processing, Springer, 2007.
11. Stinson D.R.: Cryptography: Theory and Practice. Chapman & Hall/CRC Press, Boca Raton, 2002.
12. Hassan Y.M.Y., Mohammed E.M.: PATFC: Novel Pseudorandom Affine Transformation – – Based Feistel – Network Cipher. IEEE International Symposium on Signal Processing and Information Technology, pp. 811-816, 2005.
13. Fischer S.: Analysis of lightweight stream ciphers. Ph.D. Thesis, Ecole Polytechnique F'ed'erale de Lausanne, Lausanne, 2008.
14. Shannon C.E.: Communication Theory of Secrecy Systems. Bell System Technical Journal, vol. 28, no. 4, pp. 656-715, 1949.
15. Barak B.: Lecture 2 – Perfect Secrecy and its Limitations, 2009.
16. http://www.cs.princeton.edu/courses/archive/fall05/cos433/lec2.pdf.
17. Shull R.: Cryptography, 2004.
18. http://cs.wellesley.edu/~crypto/lectures/tr05.pdf.
19. Arora S., Barak B.: Cryptography. Computational Complexity: A Modern Approach. Cambridge University Press, New Jersey, 2009.
20. Comer D.E.: Internetworking with TCP/IP. Prentice Hall, Upper Saddle River, 2005.
21. Zieliński T.P.: Cyfrowe przetwarzanie sygnałów: Od teorii do zastosowań. Wydawnictwa Komunikacji i Łączności, Warszawa, 2007. (not available in English).
22. Gajda J.: Statystyczna analiza danych pomiarowych. WEAliE AGH, Kraków, 2002. (not available in English).

Manuscript submitted 20.06.2011

STATYSTYKA W WYKRYWANIU INFORMACJI SZYFROWANEJ

G. GANCARCZYK,
A. DĄBROWSKA–BORUCH, K. WIATR

STRESZCZENIE *Nowoczesna kryptografia wykorzystuje wyszukane i skomplikowane obliczeniowo przekształcenia matematyczno-logiczne w celu ukrycia ważnej informacji jawnej przez osobami niepowołanymi. Przeważająca większość z nich nadal odwołuje się do postawionego w roku 1949 przez Claude'a E. Shannona postulatu, że idealnie utajniona informacja charakteryzuje się tym, że żaden z pojawiających się w niej symboli nie jest bardziej prawdopodobny niż inne spośród używanego alfabetu znaków.*

Zgodnie z tą definicją dane idealnie zaszyfrowane w swej naturze przypominają dane losowe o rozkładzie równomiernym, czyli przypomina swoim rozkładem szum biały.

Koncepcja detektora opiera się o algorytm analizujący podawane na wejściu dane pod względem ich podobieństwa do szumu białego. Wielkości odniesienia są bardzo dobrze znane, a ich ewentualne wyprowadzenie nie przysparza żadnych trudności. Wyznaczając w sposób doświadczalny granice tolerancji dla każdego z parametrów uzyskuje się w pełni działający algorytm, dokonujący w sposób zero–jedynkowy klasyfikacji na jawny/tajny.

W grupie przedstawionych 14 Parametrów Statystycznych pojawiają się takie jak: energia, wartość średnia czy też momenty centralne. Na ich podstawie można stworzyć klasyfikator pierwszego poziomu. Efektywność poprawnego rozróżnienia danych przez klasyfikator pierwszego rzędu waha się w granicach od 80% do 90% (w zależności od użytej w algorytmie wielkości).

W celu zwiększenia wykrywalności danych proponuje się, a następnie przedstawia, klasyfikator drugiego rzędu, bazujący na dwóch lub więcej, wzajemnie nieskorelowanych Parametrach Statystycznych. Rozwiązanie takie powoduje wzrost sprawności do około 95%.

Zaproponowany w artykule algorytm może być wykorzystany na potrzeby kryptoanalizy, statystycznej analizy danych, analizy danych sieciowych.

W artykule przedstawiona jest także koncepcja klasyfikatora trzeciego rzędu, wykorzystującego dodatkowo informacje o charakterze innym niż statystyczny, na potrzeby prawidłowej detekcji danych zaszyfrowanych.

Grzegorz GANCARCZYK, M.Sc. – AGH University of Science and Technology, Kraków, Poland, Master of Science in Electronics (2009). Employed in Academic Computer Centre CYFRONET AGH (2009 – till now).
Research interests – statistics, probability theory, stochastic processes, noise, cryptography, digital data processing, digital signal processing, hardware acceleration of numerical methods.

Agnieszka DĄBROWSKA-BORUCH, Ph.D. – AGH UST, Kraków, Poland, Master of Science in Electronics (2002), Doctor of Philosophy in Electronics (2007). Employed in Department of Electronics at AGH UST as assistant professor. Member of the Team of Acceleration at ACC CYFRONET AGH.
Research interests – image compression, real time systems, reconfigurable systems and devices.

Professor Kazimierz WIATR – AGH UST, Kraków, Poland, Master of Science in Electronics (1980), Doctor of Philosophy in Electronics (1987), Ph.D. with habilitation (1999), Professor (2002). Employed in Department of Electronics at AGH UST as professor. Director of the ACC CYFRONET AGH.
Research interests – machine vision, multiprocessor systems, reconfigurable devices, reconfigurable computing and hardware methods of calculations acceleration.