CRYPTOGRAPHIC REQUIREMENTS OF BOOLEAN FUNCTIONS

by

Aliyu DanladiHina, Abdullahi M. Auwal, & Rakiya M.K. Adamu and Bala Umar
Basic Studies Department, Federal Polytechnic, Bauchi, Computer science Department, Federal Polytechnic, Bauchi, General Studies Department, Federal Polytechnic, Bauchi
auwal2gga@yahoo.com

Abstract
Boolean functions are the building blocks of symmetric cryptographic algorithms. Symmetrical cryptographic algorithms are fundamental tools in the design of all types of digital security systems. Cryptographic applications of Boolean functions are meant to have some cryptographic properties, those properties are built to thwart cryptanalysis of certain kinds, and multiple crypto-graphic properties are usually required for a Boolean function to be used in cryptographic algorithm design, expected to resist some known attacks. Nonlinear Boolean functions are considered for a long time to construct symmetric cryptosystems. In order to resist the known attacks, many properties of Boolean functions must be utilized. In this paper we analyse some major properties according to different attacks. Therefore, the primary applications of cryptographic Boolean functions are the design of cryptographic algorithms, particularly stream cipher and block cipher algorithms. We discussed some applications of Boolean functions with cryptographic properties, where the involved Boolean functions are primary building blocks.

Keyword*: Boolean functions, symmetric cryptosystems, stream cipher and block cipher algorithms*

1. Introduction
Cryptography, originally from the Greek words 'kryptos = to hide' and 'graphein = to write' is the combination of all processes aimed at denying an intruder called an adversary, to make meaning of a message sent through an unsecure channel. Prior to the modern age, it was synonymous with encryption, the conversion of information from a readable state to apparent nonsense. Encryption does not itself prevent interference, but denies the intelligible content to the adversary. Plaintext messages (M) are encrypted into unintelligible content called ciphertext (C)using an encryption key (K_e). The original message (M) is recovered back through a process called decryption using a decryption key(K_d). [18].

$K_e(M) = C$; and $M = K_d(C)$	(1)

Cryptography could be symmetric or asymmetric. With symmetric cryptography, two separate keys are used for encryption and decryption $K_e \neq K_d$ (Fig. 1a), example of which includes the commonly used AES (Advanced Encryption Standard) which replaced the older DES (Data Encryption Standard). In an event where both encryption and decryption are done with a single key$K_e = K_d = k$,then we have what is called a asymmetric cryptography (Fig. 1b), e.g. RSA (Rivest-Shamir-Adleman), and ECC (Elliptic Curve Cryptography).

T he security of cryptographic algorithms is about successful transmission of the information without it being intercepted by an adversary. The security largely a function of the encryption key/keys. This is supported by the Kirchhoff 's principle, which states that: a cryptosystem should be secure even if everything about the system, except the key, is public knowledge [17]. The idea is that, if any part of a cryptosystem (except the individual secret key) has to be kept secret then the cryptosystem is indeed not secure. That's because if the simple act of disclosing some detail of the system were to make it suddenly insecure then

you've got a problem on your hands. The generation of the key stream is thus an issue of concern. In a stream cipher, the keystream, is the sequence which is combined, digit by digit, to the plaintext sequence for obtaining the cipher-text sequence $C = M$ The keystream is generated by a nite state automaton called the keystream generator. keystreams are required to pass prescribed statistical tests. The statistical analysis of the random sequences (keystream) is very important but alongside the application of statistical tests that assess the outcome of a randomness generator, there must be a serious analysis of the source the generator extracts randomness from [10]. The quality of a keystream by way of its randomness/complexity, largely depends on the nonlinearity of the generating function used [4]. One function that has stand out as an efficient tool in this regard is the Boolean function.

The study of Boolean functions has been a branch of cryptography for many decades. In 1949 Shannon established the foundations of modern cryptography by formulating the notion of product ciphers which use two basic cryptographic transformations: permutations and substitutions. These transformationsuse Boolean functions with desirable cryptographic properties [8, 2].

Boolean functions are the building blocks of symmetric cryptographic systems [6]. Symmetrical cryptographic algorithms are fundamental tools in the design of all types of digital security systems. A concise reference on how Boolean functions are used in cryptography. Currently, practitioners who need to apply Boolean functions in the design of cryptographic algorithms and protocols need to insight into detailed properties and/or characteristics of Boolean functions.

The most important part of a stream cipher is the key stream generator, which provides the overall security for stream ciphers. Nonlinear Boolean functions were preferred for a long time to construct the key stream generator. In order to resist several known attacks, many requirements have been proposed on the Boolean functions [7]. Attacks against the cryptosystems have forced deep research on Boolean function to allow us a more secure encryption.

Symmetric cryptosystems' security is strongly influenced by the Boolean functions deployed. Hence, the knowledge about cryptographic properties of Boolean functions will hence guide the systems' designer with regards the choice of the most suitable Boolean function to use. With a cryptographic mind, one needs to have the following knowledge about Boolean functions:

The properties of Boolean functions;

The design and implementation of Boolean functions;

The existence, distribution, construction of Boolean functions with certain properties;

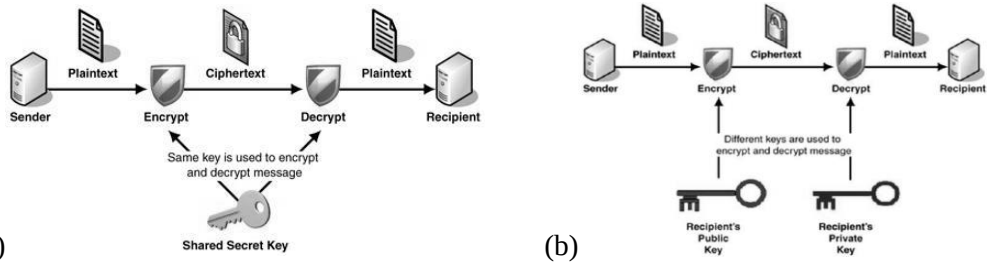


Figure 1: Cryptographic Algorithms.

The trade-off between various required properties of Boolean functions, with a view to maximize quality and improve the performance of cryptosystems;

The study of new properties according to new attacksthatwill be emerging with time.

2. Preliminaries

A Boolean function f of $n$variables is a set of all possible mappings $f: \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ where $\mathbb{F}_{2^n}$ is a Galois field of order n and $\mathbb{F}_2^n$ is its corresponding vector space [19]. Such a Boolean function can be represented as a polynomial viz:

$$f(x) = f(x_1, x_2, \dots, x_n) = \oplus_{u \epsilon \mathbb{F}_2^n} a^u X^u \qquad (2)$$

where$a \epsilon \mathbb{F}_2$ and $x, u \epsilon \mathbb{F}_2^n$. This representation is known as algebraic normal form (ANF) and its degree is called the algebraic degree.

The field of all Boolean functions B_n of order n has cardinality 2^{2n}. A Boolean function $f(x) = f(x_1, x_2, \dots, x_n)$ of n variables is usually represented by its truth table, of dimension 2^n The number of 1s in the truth table (TT) is called the Hamming weight $w_H(f)$, of the function f.

$w_H(f) = \{X: f(x) = 1, X \in \mathbb{F}_2\}$	(3)

A Boolean function is said to be balanced if $w_H(f) = 2^{n-1}$. Given two Boolean functions $f, g \in B_n$, their Hamming distance is given by $d_H(f, g) = w_H(f \oplus g)$.

$d_H(f, g) = w_H(f x_2 g) = \{X \in \mathbb{F}_2^n / f(X) \neq g(Xx)\}$	(4)

The desirable cryptographic properties of Boolean functions include among others, nonlinearity, correlation immunity, algebraic degree, linear structure, Strict avalanche criterion, Balancedness and resiliency etc. Some of these properties can be best studied using the Walsh transform.

Let $X = (x_1, x_2, \dots, x_n), w = (w_1, w_2, \dots, w_n) \in \mathbb{F}_2^n$.

Definition 2.1 (Walsh Transform) the Walsh transform of an n-variable Boolean function $f$is an integer valued function, $W_f = \mathbb{F}_2^n \rightarrow \mathbb{R}$ defined by

$W_f(w) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x) + (w.x)}$(5)

where$(w.x) = w_1 x_1 \oplus w_2 x_2 \oplus \dots \oplus w_n x_n$.

The term $W_f(w)$ is called the Walsh coefficient of f at the point w which satisfies the Parseval's equation:

$\sum_{w \in \mathbb{F}_2^n} W_f^2(w) = 2^{2^N}$.

The set of all the Walsh coefficients is referred as the Walsh spectrum of f.

The derivative of a Boolean function f with wrt to a $d_f(a)$ is define by $d_f(a) = f(x) \oplus f(x \oplus a)$. The periodic autocorrelation function of f is a real-valued function defined on all $a \in \mathbb{F}_2^n$ given by

$$C_f(a) = \sum_{x \in \mathbb{F}_2^n} (-1)^{d_{f(a)}} (6)$$

3.Complexity Requirements of Boolean functions for Cryptographic Application

3.1 Nonlinearity

The equivalence of Nonlinearity to the principle of confusion introduced by Claude Elwood Shannon in 1949 [15] allows it to be used as a measure of complexity of Boolean functions and for measuring linear attacks involved in stream and block. These linear attacks are due to the existence of affine approximations of the Boolean functions used in the systems. The Measure of the nonlinearity of the Boolean functions determines the complexity of the system developed, the measure of this nonlinearity could either be algebraic: the degree of the ANF, or functional: the minimal distance from the function to the set of all affine functions.

Definition 3.1 (Nonlinearity). The nonlinearity N_f of a Boolean function is its minimum Hamming distance to the set of all affine functions with n-variables

$$N_f = min_{\ell_a \in A_n} d_H(f, \ell_a) \quad (7)$$

whereA_n C B_n is the set of all affine functions.

In terms of the Walsh transform

□□ = $2^{□-1} - \frac{1}{2}$□ □ □□∈□$_2^□$ |□□(□)| (8)

It is widely reported in literature that □□ ≤ $2^{□-1} - 2^{□/2-1}$

3.2 Correlation Immunity

The concept of correlation immune functions was introduced by Siegenthaler [16], they are primarily designed to resist a correlation attack.

Definition 3.2 A Boolean function □ is called □ − □□ order correlation immune iff□□(□) = *0* for all vectors □ ∈ □$_2^□$ such that *1* ≤ □□(□) ≤ □.

A balanced □ − □□ order correlation immune Boolean function □ is called □-resilient. In other words, the function □ is □ − □□ order correlation immune if □□(□) = *0* for all *1* ≤ □□(□) ≤ □.

3.3 Algebraic Immunity

An annihilator of the function □ ∈ □□ is a nonzero function □ ∈ □□ such that □ □ = *0*. Given a Boolean function □, the algebraic immunity of □, □□(□)is the minimum degree of all annihilators of □or □ + *1*. As has been reported by [5, 11, 3], □□(□) ≤ [□/*2*]. In order to resist an algebraic attack, the Boolean function □(□) should have the property that there is no non-zero Boolean function □(□) such that □(□)□(□) = □(□)□ □ □□(□)has a low algebraic degree.

$$\mathcal{AI}(f) = min\{(\deg(g(X)|f(X)g(X) = 0)\} \quad (9)$$

4. Stream Ciphers

Symmetric cryptography is split into block ciphers and stream ciphers. A stream cipher is a symmetric cipher which operates with a time-varying transformation on individual plaintext digits. Stream ciphers can encrypt plaintext messages of variable length. Fig. 2 shows a general structure of a synchronous (keystream is generated independent of the plaintext stream) stream cipher showing both encryption and decryption. The keystream k = (O_1, O_2, …)isusedtoencryptthemessageM to produce the cipher-textC_i.

$$C_i = M_i \oplus O_i \quad (10)$$

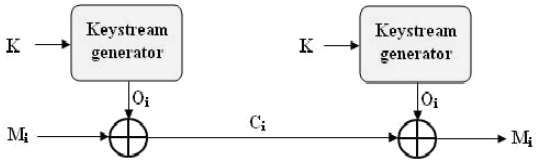


Figure 2: A stream Cipher.

The one-time pad can be thought of as an example - each message uses a portion of the key with length equal to the length of the plaintext message. (Then that portion of the key is never re-used.)

4.1 Construction Using Boolean Functions

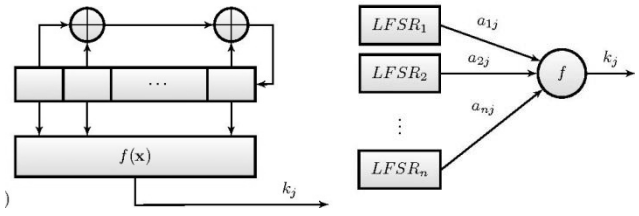
The combination of a key stream generator and an encryption algorithm makes up what is called a stream cipher [13]. The encryption algorithm XORs the plaintext with the key stream. Since the plaintext is of public knowledge, hence, the security of the stream cipher depends on the security of the key stream. The key stream generator (Fig. 3) is considered as a combination of two parts [13], the first part, referred to as the driven part consists of Linear Feedback Shift registers (LFSRs). By design, the operation of the LFSR is linear [9], thus therefore the security of the key stream generator depends on the nonlinear part, which controls the states of the generator [1, 9]. The design of the driven part is relatively simple, since the theory of LFSR is mature, especially that we have a maximum length sequence with good pseudo-randomness. The nonlinear combination part combines the sequences from the driven part into a sequence with good cryptographic properties. The use of Boolean functions in the nonlinear part of the key stream generators has been for quite some times now [12, 14]. The nonlinear Boolean functions used in the generation of the key streams could either be filter functions of combining functions. With filter functions, one LFSR is used as the driven part (Fig.3a), while the combining generator uses several LFSRs as the driven part (Fig. 3b). The corresponding key stream generators are called nonlinear filter sequence generators (see Fig.3a) and nonlinear combination sequence generators (see Fig. 3b).

4.2 Attacks and Required Properties

If we consider the nonlinear combination sequence generator (Fig. 3b), the correlation between the output sequence $\{\square_{\square}\}$ and every input sequence $\{\square_{\square\,\square}\}(1 \leq \square \leq \square)$, enables one to recover the initial states and the feedback function of $LFSR_i$ using statistical methods. This kind of attack is called correlation attack.

Another form of attack peculiar to sequence generators is the linear attack. This attacks sequence of high complexity using a sequence of low complexity. The efficiency of such an attack is proportional to the value of the Walsh transform of the Boolean function $\square_{\square}(\square)$ used.

Figure 3: Key Stream Generators



Consequently, we require a high nonlinearity of the Boolean function to resist linear attacks. In LFSR-based stream ciphers, the performance of Boolean functions against algebraic attack received considerable attention since the year 2003. Boolean functions with big algebraic immunity can resists algebraic attacks to certain extend. However, no matter how high the algebraic immunity of a Boolean function is, it may not resist a fast algebraic attack [5]. Fast algbraic attack is an improvement of algebraic attack.

In the fast algebraic attack, the Boolean function □(□) must not have a low degree multiple. The fast algebraic attack will still be efficient even if there exists a low degree (less than □□(□)) of the function □(□) such that the degree of □(□)□(□) = □(□) is not too big. If the degree of □(□)is □(□ < □/2), then fast algebraic attack can be converted to solve a system of equations with degree less than □. The research in fast algebraic attack shows that for any Boolean function □(□) andintegers □; □ such that □ + □ ≥ □, there exists Boolean function□ such that □(□)□(□) = □(□), with deg(□(□)) ≤ e and deg(□(□ □ □)) ≤ □. So there are no Boolean functions which can totally resist fast algebraic attacks, but we can choose some special Boolean functions which have high complexity against a fast algebraic attack.

Details of how the tow attacks works can be found in [5, 11] and references contained therein. Thus therefore with stream ciphers, the Boolean function used in the keystream generating algorithm should have the following properties corresponding to the kind of attack.

Statistic analysis: The Boolean function should be balanced.
Linear attack: The function used should have high nonlinearity.
Correlation attack: High correlation immunity.
Algebraic attack: High algebraic immunity.

5. Conclusions

Boolean functions are the most important part in symmetric cryptosystems. In this paper, we summarized some major properties of Boolean functions which make them resist to several attacks. One may want to find a Boolean function that holds all the security properties, but a trade-off is certainly necessary. The construction of Boolean functions with certain properties is the main research subject of cryptographic Boolean functions. We may also require new properties because attacks never stop. The quantitative evaluation of Boolean functions regarding security applications and the relation between different properties remain an important research topic.

References

Henry Beker and Fred Piper. Cipher systems: the protection of communications. Northwood Books, 1982.

Claude Carlet. Boolean functions for cryptography and error correcting codes. Boolean models and methods in mathematics, computer science, and engineering, 2:257{397, 2010.

Claude Carlet, Deepak Kumar Dalai, Kishan Chand Gupta, and SubhamoyMaitra. Algebraic immunity for cryptographically significant boolean functions: analysis and construction. IEEE Transactions on Information Theory, 52(7):3105{3121, 2006.

Henry Corrigan-Gibbs, Wendy Mu, Dan Boneh, and Bryan Ford. Ensuring high-quality randomness in

cryptographic key generation. In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pages 685{696. ACM, 2013.
Nicolas T Courtois. Fast algebraic attacks on stream ciphers with linear feedback. In Annual International Cryptology Conference, pages 176{194. Springer, 2003.
Thomas W Cusick and PantelimonStanica. Cryptographic Boolean functions and applications. Pages 7 – 23. Academic Press, 2017.
Cunsheng Ding, Guozhen Xiao, and Weijuan Shan. The stability theory of stream ciphers, volume 561. Springer Science & Business Media, 1991.
Solomon Golomb. On the classification of Boolean functions. IRE transactions on circuit theory, 6(5):176{186, 1959.
Peizhong Lu and Lianzhen Huang. A new correlation attack on LFSR sequences with high error tolerance. In Coding, Cryptography and Combinatorics, pages 67{83. Springer, 2004.
KingaMarton, AlinSuciu, and IosifIgnat. Randomness in digital cryptography: A survey. Romanian Journal of Information Science and Technology, 13(3):219{240, 2010.
Willi Meier, EnesPasalic, and Claude Carlet. Algebraic attacks and decomposition of Boolean functions. In International Conference on the Theory and Applications of Cryptographic Techniques, pages 474{491. Springer, 2004.
Willi Meier and OthmarStaffelbach. Nonlinearity criteria for cryptographic functions. In Workshop on the Theory and Application of Cryptographic Techniques, pages 549{562. Springer, 1989.
Rainer ARueppel. Stream ciphers. In Analysis and Design of Stream Ciphers, pages 5{16. Springer, 1986.
Jennifer Seberry, Xian-Mo Zhang, and YuliangZheng. On constructions and nonlinearity of correlation immune functions. In Workshop on the Theory and Application of Cryptographic Techniques, pages 181{199. Springer, 1993.
Claude Elwood Shannon. Communication in the presence of noise. Proceedings of the IRE, 37(1):10{21, 1949.
Thomas Siegenthaler. Decrypting a class of stream ciphers using ciphertext only. IEEE Transactions on computers, 1(C-34):81{85, 1985.
Nigel Paul Smart et al. Cryptography: an introduction, volume 3. McGraw-Hill New York, 2003.
Douglas R Stinson. Cryptography: theory and practice. Pages 26 – 37, Third edition, CRC press, 2005.
Zhe-Xian Wan. Lectures on finite fields and Galois rings. World Scientific Publishing Com