COLFSR - A Hybrid Random Number Generator Based on Chaos Optimisation and Linear Feedback Shift Register

Eyup Eroz[1], Erkan Tanyildizi[2], Fatih Ozkaynak[2,*]
[1]Department of Computer Technologies, Karakocan Vocational School, Firat University,
Elazig - 23119, Turkey
[2]Department of Software Engineering, Faculty of Technology, Firat University,
Elazig - 23119, Turkey
*eeroz@firat.edu.tr; etanyildizi@firat.edu.tr; *ozkaynak@firat.edu.tr*

Abstract—**Many researchers are trying to make our lives easier with developments in the Internet of Things, industry 4.0, and artificial intelligence. However, when the security of the data, which is at the centre of all these developments, is not ensured, the processes that try to make the lives of human beings more comfortable turn into nightmares. The problem that is tried to be addressed in this study is to share the details of an approach that can be used as an encryption key in hardware encrypted data storage units that can be used to address security concerns that may arise during the transmission, processing, and storage of sensitive data. The proposed method has contributed to the hybrid random number generators, both by optimising the deterministic generators and the chaotic selection algorithm. The results of the successful analysis of the proposed architecture have confirmed that it will have potential in many practical applications in the future. It is thought that with projections for future studies, it will contribute to the field of global encryption software.**

Index Terms—**Security; Hardware encryption; Randomness; Cryptographic keys; Chaos; Optimisation linear feedback shift register.**

I. INTRODUCTION

Data, which has been called the "new oil of our digital lives", now plays a key role in many areas. Therefore, the security of such a valuable asset has become increasingly important. In recent years, it has been increasingly felt that the negative effects that can be experienced in cases where data security is not given enough attention can be annoying. Many companies and organisations recognise that cyberattacks are expensive in terms of both financial losses and customer loyalty risks. Unfortunately, during the years 2020 and 2021, when the COVID epidemic deeply affected our lives, cyber threats have increasingly targeted end users who work remotely from home with the epidemic [1]–[7]. It is inevitable that attackers exploit unprepared and distracted employees with laptops full of corporate data. Considering the prediction that remote work will continue widely after the pandemic, new security precautions should be taken to adapt to this new situation. In other words, despite the changing threat environment, we now have to fight on a much broader front to defend our digital presence.

The problem that is tried to be addressed in this study is to share the details of an approach that can be used as an encryption key in hardware encrypted data storage units that can be used to address security concerns that may arise during the transmission, processing, and storage of sensitive data, as project teams use mobile and cloud platforms daily to share and store data, potentially making it accessible to attackers. In addition, data are physically moved outside companies and organisations with smartphones, removable hard drives, and USB storage devices that are prone to theft and loss. In this context, digital assets constantly cross the borders of companies, organisations, and nations. Recent research shows that there has been a 123 % increase in the volume of data downloaded to USB media by employees since the outbreak of COVID-19 [8]. These statistics show that teams are using removable storage devices to carry large volumes of data home. Unless these devices are encrypted, a spike in data breaches associated with remote working vulnerability will be inevitable in the near term. An indication that unencrypted removable storage is one of the fastest growing methods of malware entry is the seven-fold increase in ransomware cyberattacks in 2020, as stated in the BitDefender's report [9]. As a result, working online increases cybersecurity risks. To cope with these risks, encrypted hardware data storage units have been increasingly used in recent years. The most important advantage of hardware encrypted data storage units is that they can implement the encryption process without being dependent on any system since they contain a processor.

A. Definition of a Problem

The role of encryption keys used in encrypted storage units is to ensure the security of data and prevent unauthorised access. These keys are used to encrypt data and decrypt it when necessary. Choosing and managing the right keys is a critical problem to ensure that data are stored securely and protected against unauthorised access. The security of

Manuscript received 30 July, 2024; accepted 28 November, 2024.

This work was supported in part by The Scientific and Technological Research Council of Turkey under Grants Nos. 121E600 and 122E337, and Fatih Özkaynak was supported in part by Firat University Scientific Research Projects Unit (FUBAP) under Grant No. TEKF.24.27.

cryptographic keys depends on these keys being sufficiently random and of high entropy. However, achieving true randomness is quite difficult. Most computer systems use random number generators that operate with deterministic processes, which imply a certain pattern or predictability. Therefore, generating keys that are sufficiently random and unpredictable is a significant challenge. Inadequate or poor management of randomness sources increases the crackability of keys. The length of keys has a direct impact on the security of cryptographic algorithms. Longer keys are generally considered more secure; however, generating and managing long keys is also more complex. Especially in symmetric encryption methods, very long keys can be impractical and cause performance problems. Establishing this balance is a significant problem in cryptographic key generation. Cryptographic key generation usually requires complex mathematical and algorithmic processes. These processes operate under certain hardware and software limitations. These processes are difficult to execute efficiently, especially on portable devices that require low power consumption. Hardware randomness sources are generally considered more reliable; however, these sources can cause problems in practice due to their limited or costly nature. In this work, a hybrid approach is proposed to address these problems.

B. *Contribution of Paper*

The original aspect of this study is that an approach is proposed that can be used as an encryption key in hardware encrypted data storage units. The keys to be used as encryption keys must meet some requirements. A hybrid approach has been chosen to best meet these requirements. Thanks to the hybrid approach, the advantages of deterministic and true random number generators are combined in a robust architecture. The linear feedback shift register (LFSR) structure is used as a deterministic generator. LFSR structures have been studied in the literature for many years. When generating random numbers with LFSR structures, one of the most critical steps is to decide on the appropriate configuration because the length of nonrepeat random bits that can be obtained in an LFSR architecture consisting of n flip-flops is a maximum of n. However, each of the possible feedback configurations does not produce a maximum-length random bit sequence. Although configurations to generate random bits of various order maximum length have been studied in the literature, the most important contribution of this study to the current literature is to decide on the configuration to generate a maximum length sequence for any order LFSR structure through heuristic optimisation algorithms. As the degree of LFSR structure increases, the number of possible configurations, in other words, the search space will be very large, so deciding on the appropriate configuration becomes an NP problem. This problem is solved with the binary bat optimisation algorithm. The maximum length of random bits is chosen as the objective function of the optimisation algorithm.

Another important contribution of the study is the chaos-based approach proposed to improve the statistical properties of the random sequences (cryptographic keys) generated using the configurations determined by the binary bat algorithm. When analysing that a cryptological key generator has good statistical properties, it is expected to pass all tests in the 800-22 statistical test suite consisting of 15 tests offered as a standard by NIST. 1,000,000 bits must be obtained from the random number generator to perform these tests. To obtain 1,000,000 bits with the LFSR structure, a configuration consisting of 20 flip-flops must be established. 435 different configurations have been identified that can produce a maximum length sequence using the binary bat optimisation algorithm. However, none of these configurations passed all tests in the NIST 800-22 statistical test suite [10]. Another unique aspect of the study is the fact that the new sequences obtained using the chaotic mixing process, which is the innovative aspect of the study, have been shown to pass the NIST 800-22 statistical tests successfully and turn the generator into an unpredictable structure due to its excessive dependence on the initial conditions and control parameters of the chaotic system. It is thought that these original contributions and the successful analysis results will reveal many potential applications, especially hardware encrypted data storage units, and various new research questions.

C. *Organisation of Paper*

To explain all these original contributions in detail, in Section II, basic information that can guide the reader about the three main topics that form the background of the study is presented. First, the basic requirements for cryptographic randomness are defined. Then, the general working principle of the LFSR structure is explained and its advantages and disadvantages are discussed within the scope of randomness requirements. The role of optimisation algorithms and chaotic systems to overcome these disadvantages, in other words, the original aspect of the study, is discussed at the end of the second section. In Section III, the details of the proposed hybrid generator based on chaotic systems, the binary bat optimisation algorithm, and LFSR structures are explained. In Section IV, the analysis results are given, and the success of the hybrid generator is shown comparatively. In Section V, the results obtained are discussed by making a critical interpretation. In the last section, the study is summarised and a projection for future studies is presented.

II. THEORETICAL BACKGROUND

The science of cryptology is interdisciplinary in nature. It is directly related to many basic theoretical subjects of mathematics, electrical-electronic engineering, and computer sciences. To present this relationship in a systematic structure, the main components of the original proposed approach are briefly mentioned in this section, and the usage scenarios in the proposed method are explained in Section III.

The concept of randomness affects many branches of science. However, when it comes to cryptological applications, much more care should be taken. When evaluating the security of an encryption algorithm, the basic assumption is that the attacker has all the information about the system except the encryption key and has the maximum computational ability. In other words, to minimise the negative impact of the attacker, the keys to be used must meet some requirements [11]–[19], as follows.

– (R1) The encryption key must not contain statistical vulnerability. Key values should show a uniform distribution, and this statistical quality should be confirmed by various hypothesis tests.

– (R2) It should be shown that the before or after key sequence is unpredictable, even assuming that the attacker has part of the encryption key.

The main goal of the proposed method in the study is to meet these two requirements. Since meeting these two requirements is a very general problem, there is intense interest in this subject in the literature. As stated before, since randomness is one of the main problems that researchers need to address in many branches of science, independent studies have been carried out to meet both requirements. While deterministic random number generators (DRNG) [11] are generally used to meet the first requirement, true random number generators (TRNG) [12] are used for the second requirement. Since randomness has a wide spectrum of applications, both DRNG and TRNG have had application-specific successes. However, in order for these generator families to be used effectively in cryptology applications, their disadvantages should be considered as well as their advantages. In the proposed method, LFSR and heuristic optimisation algorithms are used to meet the first requirement, and chaotic systems to meet the second requirement.

Random sequences with good statistical properties, i.e., uniformly distributed, are needed in games, simulations, and many scientific calculations. Therefore, many DRNG structures have been proposed, such as linear congruential generators, LFSR and generalised feedback shift registers, Blum Blum Shub, Mersenne Twister, xorshift-based generators, and PCG [20], [21]. Another name for DRNG is pseudorandom generators. The most important reason why this generator family is called pseudorandom is that random numbers are obtained with the help of an algorithm. Since the algorithms are deterministic in nature, the outputs of the generators are not truly random. In other words, DRNG structures approximate some of the properties of real random number sequences. Although this is seen as a disadvantage, they are inexpensive as they do not require dedicated hardware and are preferred in many applications because they produce fast results. Another advantage of DRNG structures is that they produce the same sequence when starting from the same seed values. Although this jeopardises the R2 requirement, it is sometimes preferred because different applications need different requirements.

In this study, the LFSR structure was preferred due to its simple structure. LSFR is a unit with a hardware counterpart. It is obtained by connecting the flip-flop structures, which have a one-bit storage capacity, in a linear structure. It consists of one unit of shift bits with each clock cycle. The difference from normal shift registers is that the number to be added from the left is not 0 or 1 by default; It is due to the fact that it is created with a certain mathematical formula.

A schematic representation of an example LFSR structure is given in Fig. 1. When a four-bit LFSR in the example is clocked, it starts shifting from bit S0 and all bits in turn shift right one bit. The value of the S4 bit is calculated according to (1) and written to the leftmost bit value

$$s4 = c3\times s3 + c2\times s2 + c1\times s1 + c0\times s0 \left(\operatorname{mod} 2\right). \qquad (1)$$

Since the product expression here is in mod 2 arithmetic, it can be described with an AND gate. Likewise, the sum expression can be expressed with the XOR gate. LFSRs work like a state machine. They switch to a different state with each clock pulse. As a special case, LFSRs cannot start from all bits being zero. If all bits are zero, the LFSR remains constant in a single state. Therefore, after LFSR starts from any state and goes to all the states it can go to, it returns to the state it started again. The main point here is that the LFSR should go to as many states as possible with the number of bits it has. Thus, it can work more efficiently. For example, a four-bit LFSR in Fig. 1 can switch to a maximum of 2^4 - 1 = 15 different states. LFSRs in this situation are called “maximal period LFSRs”. In other words, they have the ability to visit all states in state space [22]–[25].

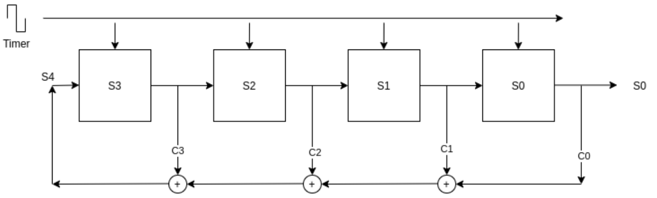


Fig. 1. An example LFSR structure.

The c values of the LFSR can be expressed as the coefficients of a polynomial. The coefficients of this polynomial are defined in GF(2). For an LFSR to be maximal, the link polynomial must be primitive. One of the objectives of this study is to obtain combinations of LFSR with maximal period. However, this is a difficult problem. Because if the generator uses a fixed amount of memory, after a sufficient number of loop steps, it will come to the same internal state the second time, and then it will repeat forever. The amount of memory (the number of flip-flops) can be increased to have a wider period. With each memory bit added, the maximum period will double. But the main question to be answered in this case is the statistical quality of the output of the pseudorandom number generator. To be able to test this, in other words, to control the R1 requirement, bit strings with a length of 1,000,0000 will be needed. However, it is an NP problem to determine which c values will be feedback in an LFSR design with n flip-flops. To solve this problem, heuristic optimisation algorithms are used in the proposed approach.

The bat algorithm is an important optimisation algorithm in the literature. Successful optimisation algorithm returns 0/1, Yes/No, etc. It has been made suitable for binary optimisation problems to provide suitable solutions to problems in the form [26]–[29]. The binary search space can be thought of as a hypercube. The search agents (particles) of the binary optimisation algorithm can flip a varying number of bits, shifting them only to the closer and farther corners of this hypercube [30]. Therefore, the velocity and position update equations in the binary bat algorithm must be modified to suit the binary search space. To update the positions from 0 to 1 or vice versa in the binary search space, the design should be made so that there is a connection between the velocity and the position update. In the discrete binary space, position update means switching between 0 and 1 values, and this must be done according to the speed of the search agents. A transfer function is required to map velocity values to probability values to update positions. This transfer function allows particles to move in binary space. The concepts of Rashedi, Nezamabadi-pour, and Saryazdi should be considered when selecting a transfer function to match velocity values with probability values [31], [32].

We mentioned that there are two basic requirements for cryptological randomness. In order to provide the condition (R1 requirement) of showing good statistical properties from these requirements; In other words, it has been explained in the previous two sections that the problem of finding configurations that can pass statistical tests among LFSR designs that will have a period of 1,000,000-bit-length can be solved by using heuristic optimisation algorithms in the proposed method. However, since the obtained random number sequences have a deterministic structure, additional measures must be taken to be used for cryptological purposes, in other words, to meet the R2 requirement. To meet this requirement, chaos theory will be used in the proposed method.

Chaos theory has a wide range of applications, from the motion of planets and meteorites to electronic systems, from meteorology to climate forecasts, stock market and economy, from earthquakes to explaining the behaviour of subatomic particles [33]. For this reason, this theory, which combines many branches of science and provides interdisciplinary work, has a very important place in science [34]. In physics, we often use models to explain phenomena and events. These models give us measurable and forward-looking mathematical and meaningful results about the system. Sometimes these results can be quite complex. So what we call chaos is the term used to describe the apparent complex behaviour of phenomena/modellings that we see as simple and well-executed systems [33].

In the 1960s, the mathematician Edward Lorenz, while working as a meteorologist at the Massachusetts Institute of Technology, showed that a small change in the initial conditions of a system would have major and unpredictable consequences in the evolution of the system [35]. In 1975, biologist Robert May showed that chaos theory is also in biology. When he examined the number of biological populations in a system over time, he determined that chaoticity occurs under certain conditions. These phenomena are explained with logistic maps and iteration graphs [34]–[36].

Since it is one of the first studies, the logistic map was used in this study due to its widespread use in the literature and its simple structure [36]. In fact, there are various studies in the literature that point to various problems of the logistic map as a chaotic system [37]. Our aim is to show that the proposed method can produce successful results even in the simplest chaotic system.

III. PROPOSED METHOD

The process of obtaining random number sequences requires serious and careful mathematical analysis. To draw attention to this importance, John von Neumann's words "Anyone who tries to generate random numbers by arithmetic methods is committing a grave sin" and Robert R. Coveyou's words "The generation of random numbers is too important to be performed randomly" are warnings frequently encountered by researchers who will start working on this subject [38]. Because pseudorandom numbers cover an important part of modern computing, they are used in many places, from cryptology to Monte Carlo methods for simulating physical systems. However, in practice, there are some situations that prevent many pseudorandom number generators from passing statistically significant tests. Some of those:

- Shorter than expected periods for some seed (initial) values;
- Poor dimensional distribution;
- Consecutive values are not independent;
- Some bits may be "more random" than others;
- Lack of uniformity.

The aim of the method proposed in this study is to develop a theoretical approach that can eliminate these problems. The general structure of the proposed architecture is presented in Fig. 2.

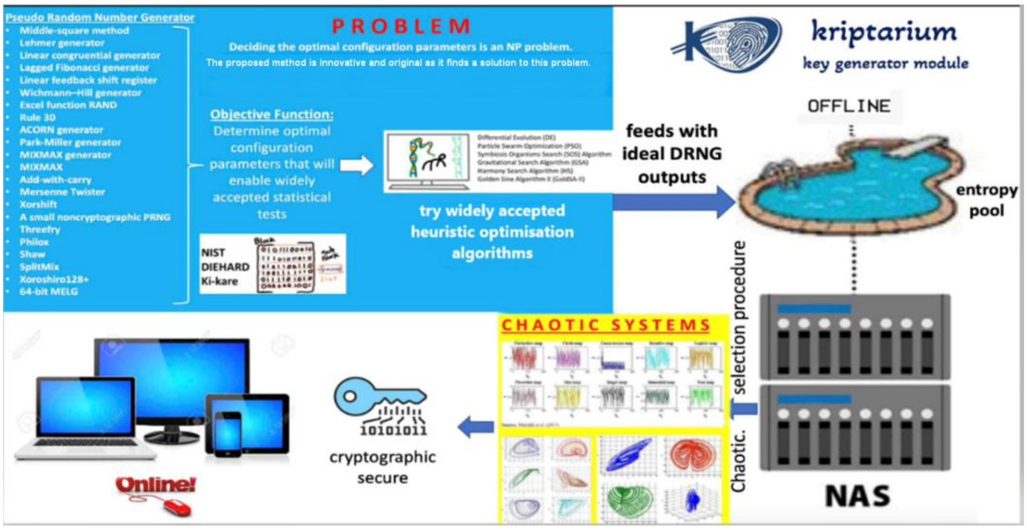

Pseudo Random Number Generator
Middle-square method
Lehmer generator
Linear congruential generator
Lagged Fibonacci generator
Linear feedback shift register
Wichmann–Hill generator
Excel function RAND
Rule 30
ACORN generator
Park-Miller generator
MIXMAX generator
MIXMAX
Add-with-carry
Mersenne Twister
Xorshift
A small noncryptographic PRNG
Threefry
Philox
Shaw
SplitMix
Xoroshiro128+
64-bit MELG
PROBLEM
Deciding the optimal configuration parameters is an NP problem.
The proposed method is innovative and original as it finds a solution to this problem.
Objective Function:
Determine optimal configuration parameters that will enable widely accepted statistical tests
Differential Evolution (DE)
Particle Swarm Optimization (PSO)
Symbiosis Organisms Search (SOS) Algorithm
Gravitational Search Algorithm (GSA)
Harmony Search Algorithm (HS)
Golden Sine Algorithm II (GoldSA-II)
NIST
DIEHARD
Ki-kare
try widely accepted heuristic optimisation algorithms
kriptarium
key generator module
OFFLINE
feeds with ideal DRNG outputs
entropy pool
Chaotic. selection procedure
CHAOTIC SYSTEMS
NAS
cryptographic secure
Online!

Fig. 2. Overview of the proposed hybrid random number generator.

The hybrid generator proposed in Fig. 2 consists of two main parts. The blue block forms the deterministic leg of the generator. The blue block can be based on any DRNG structure as input. More than twenty PRNG structures are listed for reference in the figure. In fact, there are many more PRNG structures than listed here. Some of the most wellknown are included only as projections for future studies. To show the workings of the theoretical approach on an example, the LFSR structure was used as the DRNG structure in the analyses. In the second stage of the blue block, the heuristic optimisation algorithm comes into play. Optimisation is the work done to obtain maximum output from a system within certain constraints. Here, our aim is to decide on the LFSR feedback polynomial coefficients and seed values that can pass the statistical tests successfully. The NIST test was used as the objective function in the optimisation algorithm. This test is a statistical test that is generally accepted in the literature. It examines the statistical properties of the random sequence from different angles with 15 different hypothesis tests. Bit strings with a length of at least 1,000,000 are needed to perform these tests. Therefore, the LFSR structure consists of 20 flip-flops. LFSR outputs showing good statistical properties feed an entropy pool. To allow these outputs obtained as a result of R&D studies to be used practically in real-world applications, they were transformed into a commercial product in cooperation with the Kriptarium company [39]. For this reason, the entropy pool is stored in the network-attached storage (NAS) unit. A NAS is a file-level computer data storage server connected to a computer network that provides data access to a heterogeneous group of clients.

The first unique aspect of the proposed method is to obtain configurations that can pass statistical tests through DRNG structure optimisation algorithms, while the other unique aspect is the chaotic selection procedure. Thanks to this developed selection procedure, the R2 requirement is addressed. In Fig. 2, this process is tried to be represented in the yellow block. As in the blue block, all alternatives (discrete-time, continuous-time, hyperchaotic, etc.) that can be used here are tried to be represented. On the practical side, it is possible for a client logging into the system with user name, password, and time information to obtain a key sequence of the required length from the entropy pool by determining the initial conditions of the selected chaotic system. Figure 2 can represent the operation of the system in practical applications.

The steps of the key generator algorithm are as follows.

Step 1. The end user logs into the system.

Step 2. The user name, password, time information, biometric features, or behavioural features can be used during login according to the security level required by the application.

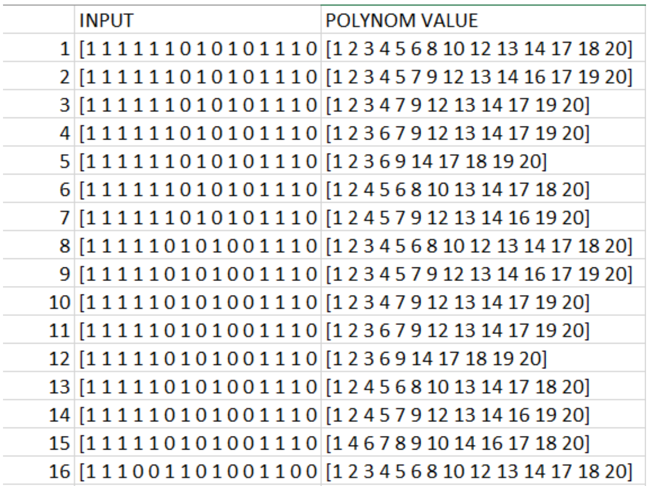
Step 3. With the help of software as a service (SaaS) offered by Kriptarium, user data are normalised as the initial condition of the chaotic system and other system parameters.

Step 4. A bit string of the length requested by the user is selected from the entropy pool via the chaotic system output.

Step 5. Cryptological keys are returned in the format requested by the user.

IV. ANALYSIS RESULTS

In this section, we try to test the success of the outputs of the proposed method. 1,000,000-bit-length sequences are needed to measure the quality of the random number sequence. An LFSR architecture consisting of 20 flip/flop structures is needed to produce outputs that will have 1,000,000-bit periods with the LFSR structure. With the optimisation algorithms, the feedback polynomial coefficients and seed values that will produce the outputs with the maximum period are calculated. As a result of the analysis, 435 different combinations were obtained. NIST statistical tests were performed for each of these 435 combinations. All of these results have been publicly shared to form a basis for future studies. It is expected that both the configurations obtained through the optimisation algorithms and the results of the NIST statistical tests will create a strong motivation for many studies in the future [39]. Hardware equipment we used in the study: Intel (R) Core (TM) i7-10750H CPU @ 2.60 GHz 2.59 GHz processor, 32 GB RAM Monster brand Tulpar 7 series laptop. Figure 3 shows the general view of the shared data set.



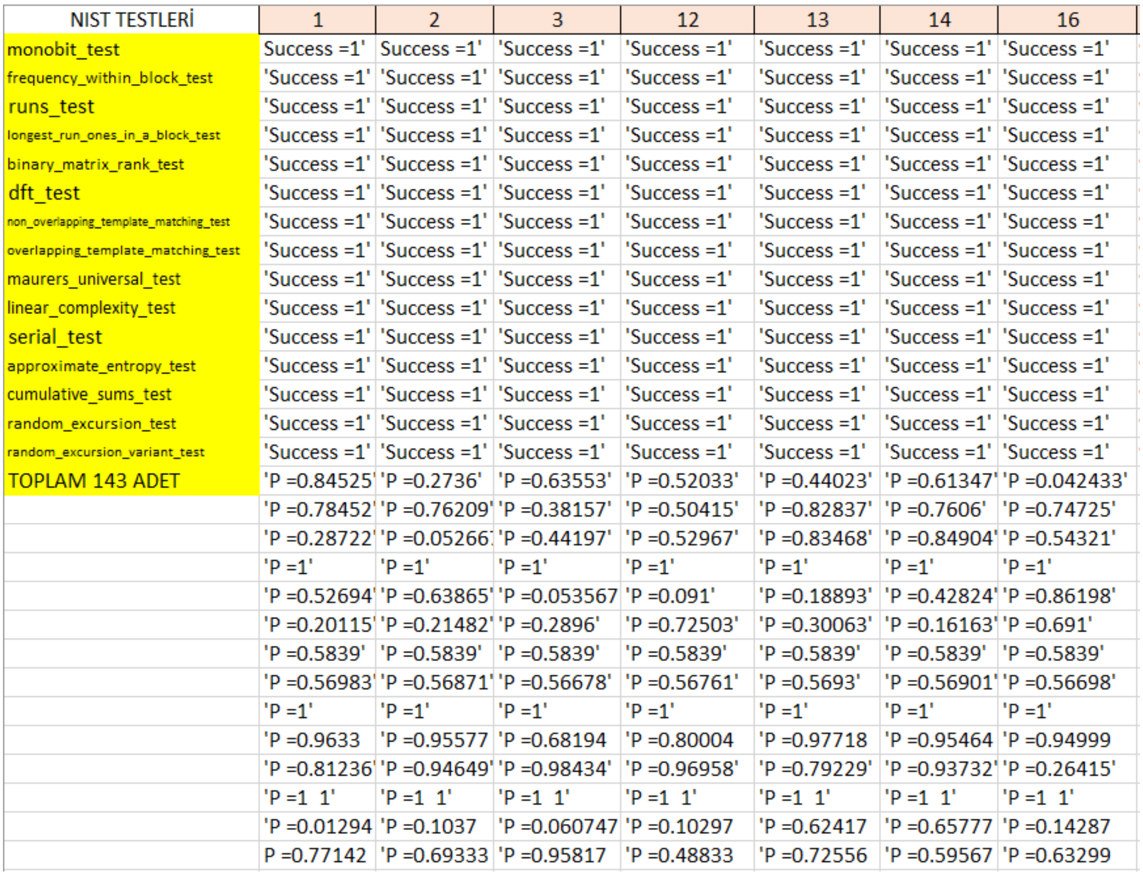
	INPUT	POLYNOM VALUE
1	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 3 4 5 6 8 10 12 13 14 17 18 20]
2	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 3 4 5 7 9 12 13 14 16 17 19 20]
3	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 3 4 7 9 12 13 14 17 19 20]
4	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 3 6 7 9 12 13 14 17 19 20]
5	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 3 6 9 14 17 18 19 20]
6	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 4 5 6 8 10 13 14 17 18 20]
7	[1 1 1 1 1 1 0 1 0 1 0 1 1 1 0	[1 2 4 5 7 9 12 13 14 16 19 20]
8	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 3 4 5 6 8 10 12 13 14 17 18 20]
9	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 3 4 5 7 9 12 13 14 16 17 19 20]
10	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 3 4 7 9 12 13 14 17 19 20]
11	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 3 6 7 9 12 13 14 17 19 20]
12	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 3 6 9 14 17 18 19 20]
13	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 4 5 6 8 10 13 14 17 18 20]
14	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 2 4 5 7 9 12 13 14 16 19 20]
15	[1 1 1 1 1 0 1 0 1 0 0 1 1 1 0	[1 4 6 7 8 9 10 14 16 17 18 20]
16	[1 1 1 0 0 1 1 0 1 0 0 1 1 0 0	[1 2 3 4 5 6 8 10 12 13 14 17 18 20]

Fig. 3. Overview of the publicly shared data set.

The data set shared in accordance with the open data policy is presented as an excel file. This file contains four pages. It contains the configurations obtained with the help of the given optimisation algorithm presented on the first page. There are three columns on this page. The configuration of the first column represents the sequence number, the second column represents the seed value, and the third column represents the feedback polynomial coefficients. Figure 4 presents to explain the structure of the remaining three pages of the excel file. The remaining three pages show the NIST test results for configurations that passed 15, 14, and 13 NIST tests, respectively. Here, the first column contains the NIST test and the other columns contain the NIST test results of the configurations. As NIST tests are a hypothesis test, only pass/fail results are not presented. In addition, P (probability) values are also given. Descriptions of NIST tests and details on how calculations are performed are available in [10], [40], [41]. 145 of these tests passed all tests successfully. 256 of them passed 14 tests. 31 of them were successful in 13 tests.

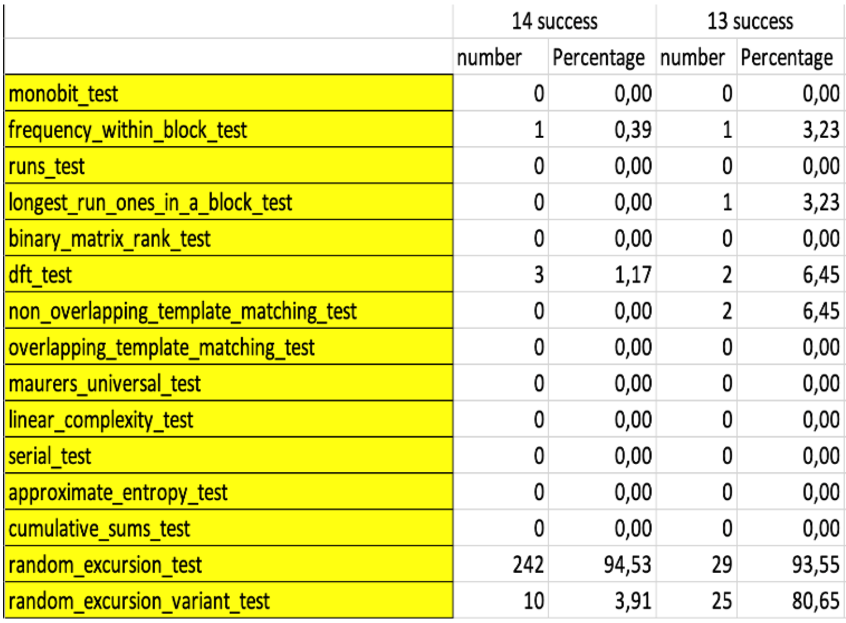
The analysis data set of the failed tests is given on the last page of the excel file. The structure of this page is shown in Fig. 5. The analysis results actually show that failure is clustered around certain tests. In other words, these hypothesis tests can be claimed to have been unsuccessful due to the generation of 1,000,000 bits.

Another widely used statistical hypothesis test as the NIST test results is chi-square analysis. In the interpretation of this analysis, it is used that the confidence value calculated for a certain degree of freedom is smaller than the acceptance value. In Fig. 6, the calculated chi-square values for each of the 143 different configurations that passed 15 NIST tests are shown in red. Green and blue values represent the break points of two different confidence values. The fact that the calculated chi-square values are below the limit values indicates that the results obtained are also safe according to the chi-square analysis. The work in [42] can be examined for further details on the calculations of the chi-square test. Another analysis approach used to check the statistical independence of the data is the autocorrelation and entropy analysis. To avoid detracting from the scope of the study, the details of these tests are not mentioned in detail. However, for a good evaluation guide on this subject, studies published by Garipcan Erdem [41] can be examined. Correlation and entropy analyses for configurations that pass the NIST tests are given in Figs. 7 and 8.



NIST TESTLERİ	1	2	3	12	13	14	16
monobit_test	Success =1'	Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
frequency_within_block_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
runs_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
longest_run_ones_in_a_block_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
binary_matrix_rank_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
dft_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
non_overlapping_template_matching_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
overlapping_template_matching_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
maurers_universal_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
linear_complexity_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
serial_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
approximate_entropy_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
cumulative_sums_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
random_excursion_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
random_excursion_variant_test	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'	'Success =1'
TOPLAM 143 ADET	'P =0.84525	'P =0.2736'	'P =0.63553'	'P =0.52033'	'P =0.44023'	'P =0.61347'	'P =0.042433'
	'P =0.78452	'P =0.76209'	'P =0.38157'	'P =0.50415'	'P =0.82837'	'P =0.7606'	'P =0.74725'
	'P =0.28722	'P =0.05266	'P =0.44197'	'P =0.52967'	'P =0.83468'	'P =0.84904'	'P =0.54321'
	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'
	'P =0.52694	'P =0.63865'	'P =0.053567	'P =0.091'	'P =0.18893'	'P =0.42824'	'P =0.86198'
	'P =0.20115	'P =0.21482'	'P =0.2896'	'P =0.72503'	'P =0.30063'	'P =0.16163'	'P =0.691'
	'P =0.5839'	'P =0.5839'	'P =0.5839'	'P =0.5839'	'P =0.5839'	'P =0.5839'	'P =0.5839'
	'P =0.56983	'P =0.56871'	'P =0.56678'	'P =0.56761'	'P =0.5693'	'P =0.56901'	'P =0.56698'
	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'	'P =1'
	'P =0.9633	'P =0.95577	'P =0.68194	'P =0.80004	'P =0.97718	'P =0.95464	'P =0.94999
	'P =0.81236	'P =0.94649'	'P =0.98434'	'P =0.96958'	'P =0.79229'	'P =0.93732'	'P =0.26415'
	'P =1 1'	'P =1 1'	'P =1 1'	'P =1 1'	'P =1 1'	'P =1 1'	'P =1 1'
	'P =0.01294	'P =0.1037	'P =0.060747	'P =0.10297	'P =0.62417	'P =0.65777	'P =0.14287
	P =0.77142	'P =0.69333	'P =0.95817	'P =0.48833	'P =0.72556	'P =0.59567	'P =0.63299

Fig. 4. NIST test results for configurations.



	14 success		13 success	
	number	Percentage	number	Percentage
monobit_test	0	0,00	0	0,00
frequency_within_block_test	1	0,39	1	3,23
runs_test	0	0,00	0	0,00
longest_run_ones_in_a_block_test	0	0,00	1	3,23
binary_matrix_rank_test	0	0,00	0	0,00
dft_test	3	1,17	2	6,45
non_overlapping_template_matching_test	0	0,00	2	6,45
overlapping_template_matching_test	0	0,00	0	0,00
maurers_universal_test	0	0,00	0	0,00
linear_complexity_test	0	0,00	0	0,00
serial_test	0	0,00	0	0,00
approximate_entropy_test	0	0,00	0	0,00
cumulative_sums_test	0	0,00	0	0,00
random_excursion_test	242	94,53	29	93,55
random_excursion_variant_test	10	3,91	25	80,65

Fig. 5. Distribution of the failed NIST test results.

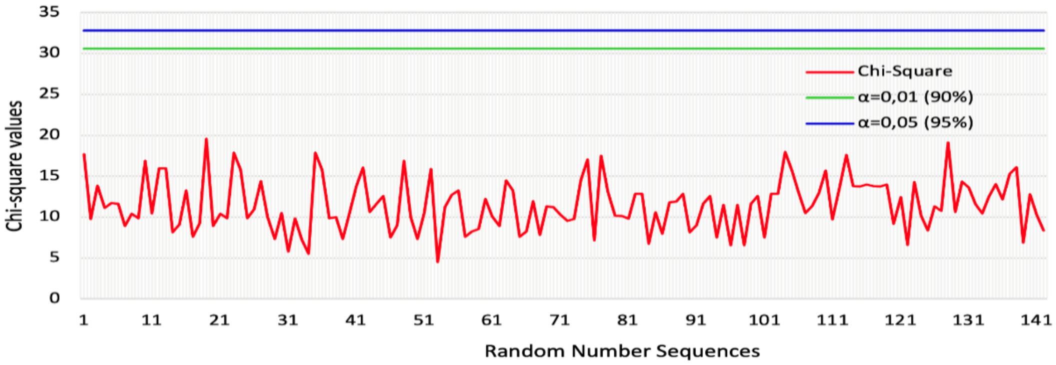


Fig. 6. Chi-square analysis results for each of the LFSR configurations determined by optimisation algorithms that pass all NIST 800-22 statistical tests.

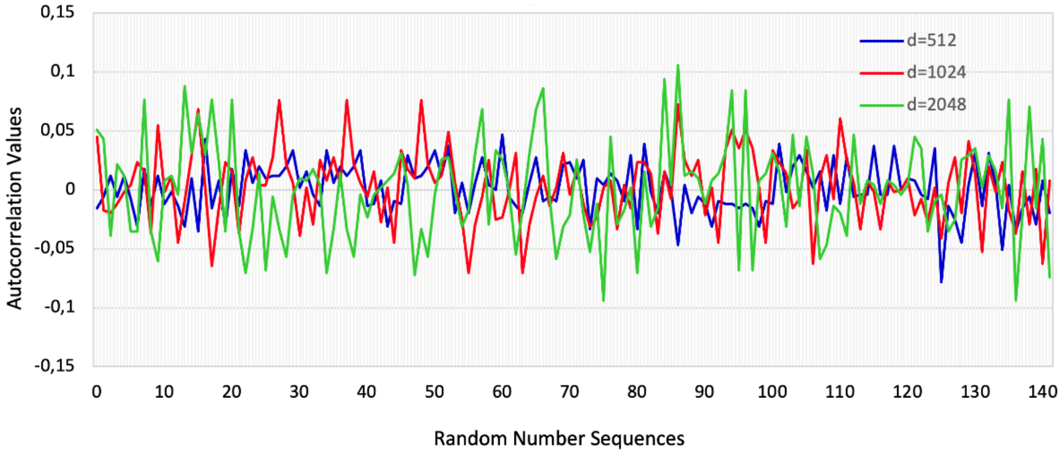


Fig. 7. Autocorrelation analysis results for each of the LFSR configurations determined by optimisation algorithms that pass all NIST 800-22 statistical tests.

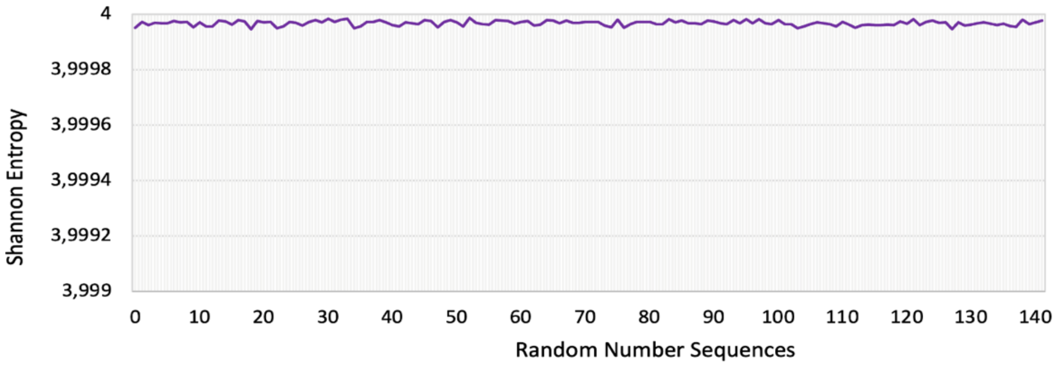


Fig. 8. Entropy analysis results for each of the LFSR configurations determined by optimisation algorithms that pass all NIST 800-22 statistical tests.

V. DISCUSSION

The proposed hybrid key generator architecture effectively combines deterministic PRNGs with chaotic systems, leveraging the strengths of both to produce highly reliable and secure cryptographic keys. The successful passing of NIST, chi-square, autocorrelation, and entropy tests validates the robustness of the generated sequences, making this approach a viable solution for practical cryptographic applications. The shared data set and detailed analysis results provide a valuable foundation for future research and development in this area, potentially leading to more advanced and secure encryption solutions. The main findings are as follows.

– The approach used a linear feedback shift register (LFSR) architecture with 20 flip/flop structures to generate sequences of 1,000,000 bits, essential for ensuring robust statistical properties.

– Feedback polynomial coefficients and seed values were optimised to produce outputs with the maximum period, resulting in 435 distinct combinations. Each combination underwent NIST statistical tests to validate their performance.

– The results of the NIST statistical tests demonstrated that 145 of the combinations tested successfully passed all 15 NIST tests, while 256 passed 14 tests, and 31 passed 13 tests. These outcomes indicate a high level of reliability and robustness in the generated sequences.

– The clustering of failed tests around specific hypotheses suggests that these issues are primarily due to the inherent properties of the 1,000,000-bit generation process.

– The results of the chi-square analysis showed that all configurations that passed the NIST tests also fell below the chi-square limit values, indicating their statistical soundness and further reinforcing the reliability of the generated sequences.

– The autocorrelation results confirmed that the sequences maintained a low correlation essential for cryptographic applications.

VI. CONCLUSIONS

The importance of data is now recognised by all stakeholders in the society. As much as the value of data, another topic that everyone agrees on is how to address the problem of data security because users adapt very quickly to new technologies. However, in this process, because they knowingly or unknowingly use weaker security solutions, they are faced with material and moral damages caused by hackers who have begun to exploit the weaknesses of these weak security solution proposals, incomparably and ruthlessly. Therefore, the interest in encryption software is increasing day by day. Many institutions and organisations are starting to need strong encryption capabilities more than ever to strengthen their security portfolios. To put this requirement numerically, the global cryptocurrency market cap, which was $8.49 billion in 2020, has reached $10.9 billion in 2021. This value is expected to reach $22.1 billion in 2026 and $59.5 billion in 2028. Statistics show that 81 % of cyberattacks are password-based cyber threats. Another factor that triggers these vulnerabilities is related to the spread of IoT devices. Responding to a survey on global IoT trends, more than two-thirds of IT security experts said they plan to use the cloud, at least in conjunction with locally used systems, to authenticate IoT devices. As IoT devices become more common, it will become increasingly difficult to manage passwords for the people responsible for their management.

In this study, an enterprise architecture is proposed to generate cryptological keys. The proposed solution uses a hybrid key generator architecture. The deterministic leg of the proposed generator is the optimised PRNG generators. An important contribution of the study is that it is based on the principle of obtaining configuration parameters that will ideally meet the statistical tests of existing PRNG structures. In the real random leg of the generator, chaos theory was used. The unpredictable nature of chaotic systems and the needs of users were tried to be addressed by making choices from the wide entropy pool. The operation of the proposed architecture is explained step by step through sample scenarios. The successful analysis of the sequences of numbers obtained indicated that the output could be used in practical applications. Due to the shared projections for future studies, it is thought that this theoretical architecture can be used to address the need for encryption software with new studies to be carried out in the future.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] O. Cherqi, H. Hammouchi, M. Ghogho, and H. Benbrahim, “Leveraging Open Threat Exchange (OTX) to understand spatio-temporal trends of cyber threats: Covid-19 case study”, in *Proc. of 2021 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2021, pp. 1–6. DOI: 10.1109/ISI53945.2021.9624677.

[2] M. Hijji and G. Alam, “A Multivocal Literature Review on growing social engineering based cyber-attacks/threats during the COVID-19 pandemic: Challenges and prospective solutions”, *IEEE Access*, vol. 9, pp. 7152–7169, 2021. DOI: 10.1109/ACCESS.2020.3048839.

[3] J. Ahmed and Q. Tushar, “Covid-19 pandemic: A new era of cyber security threat and holistic approach to overcome”, in *Proc. of 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 2020, pp. 1–5. DOI: 10.1109/CSDE50874.2020.9411533.

[4] A. Al Shammari, R. R. Maiti, and B. Hammer, “Organizational security policy and management during Covid-19”, in *Proc. of SoutheastCon 2021*, 2021, pp. 1–4. DOI: 10.1109/SoutheastCon45413.2021.9401907.

[5] S. Hakak, W. Z. Khan, M. Imran, K.-K. R. Choo, and M. Shoaib, “Have you been a victim of COVID-19-related cyber incidents? Survey, taxonomy, and mitigation strategies”, *IEEE Access*, vol. 8, pp. 124134–124144, 2020. DOI: 10.1109/ACCESS.2020.3006172.

[6] K. Halouzka, P. Kozak, L. Burita, and P. Matoulek, “Personal cyber security in email communication”, in *Proc. of 2021 International Conference on Military Technologies (ICMT)*, 2021, pp. 1–5. DOI: 10.1109/ICMT52455.2021.9502740.

[7] F. Delerue, “Covid-19 and the cyber pandemic: A plea for international law and the rule of sovereignty in cyberspace”, in *Proc. of 2021 13th International Conference on Cyber Conflict (CyCon)*, 2021, pp. 9–24. DOI: 10.23919/CyCon51939.2021.9468306.

[8] “Comprehensive DLP for Rapid Deployment and Results”, Digital Guardian: Data Protection, FORTRA. [Online]. Available: https://digitalguardian.com/about/news-events/press-releases/digital-guardian-announces-inaugural-dg-data-trends-report-which

[9] “Global Ransomware and Cyberattacks on Healthcare Spike during Pandemic”, Bitdefender, 2020. [Online]. Available: https://www.bitdefender.la/post/global-ransomware-and-cyberattacks-on-healthcare-spike-during-pandemic

[10] L. E. Bassham *et al.*, “A statistical test suite for random and pseudorandom number generators for cryptographic applications”, *Special Publication (NIST SP) - 800-22 Rev 1a*, 2010. DOI: 10.6028/NIST.SP.800-22r1a.

[11] W. Schindler, “Evaluation criteria for physical random number generators”, in *Cryptographic Engineering*. Springer, Boston, MA, 2008, pp. 25–54. DOI: 10.1007/978-0-387-71817-0_3.

[12] W. Killmann and W. Schindler, “A proposal for: Functionality classes for random number generators”, 18 Sep. 2011.

[13] M. Sönmez Turan, E. Barker, J. Kelsey, K. A. McKay, M. L. Baish, and M. Boyle, “Recommendation for the entropy sources used for random bit generation”, *NIST Special Publication 800-90B*, 2018. DOI: 10.6028/NIST.SP.800-90B.

[14] U. Ansari, A. K. Chaudhary, and S. Verma, “Enhanced True Random Number Generator (TRNG) using sensors for IoT security applications”, in *Proc. of 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT)*, 2022, pp. 1593–1597. DOI: 10.1109/ICICICT54557.2022.9917919.

[15] K. Lee, S.-Y. Lee, C. Seo, and K. Yim, “TRNG (true random number generator) method using visible spectrum for secure communication on 5G network”, *IEEE Access*, vol. 6, pp. 12838–12847, 2018. DOI: 10.1109/ACCESS.2018.2799682.

[16] B. K. Park *et al.*, “Practical true random number generator using CMOS image sensor dark noise”, *IEEE Access*, vol. 7, pp. 91407–91413, 2019. DOI: 10.1109/ACCESS.2019.2926825.

[17] A. Dheeraj, P. Das, K. K. A, S. Kalanadhabhatta, and A. Acharyya, “Modeling attacks resilient Multiple PUF-CPRNG architecture design methodology”, in *Proc. of 2022 IEEE 35th International System-on-Chip Conference (SOCC)*, 2022, pp. 1–6. DOI: 10.1109/SOCC56010.2022.9908089.

[18] M. Garcia-Bosque, A. Perez-Resa, C. Sanchez-Azqueta, C. Aldea, and S. Celma, “Chaos-based bitwise dynamical pseudorandom number generator on FPGA”, *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 1, pp. 291–293, 2019. DOI: 10.1109/TIM.2018.2877859.

[19] S. Kalanadhabhatta, D. Kumar, K. K. Anumandla, S. A. Reddy, and A. Acharyya, “PUF-based secure chaotic random number generator design methodology”, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 7, pp. 1740–1744, 2020. DOI: 10.1109/TVLSI.2020.2979269.

[20] H. Jiang, C. Li, and J. Fan, “Research on pseudo-random characteristics of new random components”, in *Proc. of 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, 2019, pp. 163–167. DOI: 10.1109/AIAM48774.2019.00040.

[21] B. Zhang, C. Xu, and D.-G. Feng, “Design and analysis of stream ciphers: Past, present and future directions”, *Journal of Cryptologic Research*, vol. 3, no. 6, pp. 527–545, 2016. DOI: 10.13868/j.cnki.jcr.000149.

[22] Wikipedia contributors (2024, August 20), Berlekamp–Massey algorithm, Wikipedia, The Free Encyclopedia. [Online]. Available: https://en.wikipedia.org/wiki/Berlekamp%E2%80%93Massey_algorithm

[23] A. Bagalkoti, S. B. Shirol, R. S, P. Kumar, and R. B. S, "Design and implementation of 8-bit LFSR, bit-swapping LFSR and weighted random test pattern generator: A performance improvement", in *Proc. of 2019 International Conference on Intelligent Sustainable Systems (ICISS)*, 2019, pp. 82–86. DOI: 10.1109/ISS1.2019.8908063.

[24] Y. G. Praveen Kumar, B. S. Kariyappa, and M. Z. Kurian, "Implementation of power efficient 8-bit reversible linear feedback shift register for BIST", in *Proc. of 2017 International Conference on Inventive Systems and Control (ICISC)*, 2017, pp. 1–5. DOI: 10.1109/ICISC.2017.8068620.

[25] M. B, G. Remadevi, and R. Bakthavatchalu, "Design of a programmable low power linear feedback shift register for BIST applications", in *Proc. of 2022 IEEE International Test Conference India (ITC India)*, 2022, pp. 1–4. DOI: 10.1109/ITCIndia202255192.2022.9854556.

[26] F. Liu, X. Yan, and Y. Lu, "Feature selection for image steganalysis using binary bat algorithm", *IEEE Access*, vol. 8, pp. 4244–4249, 2020. DOI: 10.1109/ACCESS.2019.2963084.

[27] E. Eroz, E. Tanyildizi, and F. Ozkaynak, "Determination of suitable configuration parameters for linear feedback shift register using binary bat optimization algorithm", in *Proc. of IEEE EUROCON 2021 - 19th International Conference on Smart Technologies*, 2021, pp. 348–351. DOI: 10.1109/EUROCON52738.2021.9535616.

[28] K. Atefi, H. Hashim, and T. Khodadadi, "A hybrid anomaly classification with Deep Learning (DL) and Binary Algorithms (BA) as Optimizer in the Intrusion Detection System (IDS)", in *Proc. of 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*, 2020, pp. 29–34. DOI: 10.1109/CSPA48992.2020.9068725.

[29] W. A. H. M. Ghanem *et al.*, "Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks", *IEEE Access*, vol. 10, pp. 76318–76339, 2022. DOI: 10.1109/ACCESS.2022.3192472.

[30] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm", in *Proc. of the 1997 IEEE International Conference on Systems, Man and Cybernetics. Computational Cybernetics and Simulation*, 1997, pp. 4104–4108, vol. 5. DOI: 10.1109/ICSMC.1997.637339.

[31] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "BGSA: Binary gravitational search algorithm", *Natural Computing*, vol. 9, no. 3, pp. 727–745, 2010. DOI: 10.1007/s11047-009-9175-3.

[32] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm", *Neural Computing and Applications*, vol. 25, nos. 3–4, pp. 663–681, 2014. DOI: 10.1007/s00521-013-1525-5.

[33] S. H. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Boca Raton: CRC Press, 2019, pp. 1–513. DOI: 10.1201/9780429492563.

[34] M. S. Acikkapi and F. Ozkaynak, "A method to determine the most suitable initial conditions of chaotic map in statistical randomness applications", *IEEE Access*, vol. 9, pp. 1482–1494, 2021. DOI: 10.1109/ACCESS.2020.3046470.

[35] E. N. Lorenz, "Deterministic nonperiodic flow", *Journal of the Atmospheric Sciences*, vol. 20, no. 2, pp. 130–141, 1963. DOI: 10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2.

[36] A. Anees and I. Hussain, "A novel method to identify initial values of chaotic maps in cybersecurity", *Symmetry*, vol. 11, no. 2, p. 140, 2019. DOI: 10.3390/SYM11020140.

[37] Y. Aydin and F. Ozkaynak, "Eligibility analysis of different chaotic systems derived from logistic map for design of cryptographic components", in *Proc. of 2021 International Conference Engineering Technologies and Computer Science (EnT)*, 2021, pp. 27–31. DOI: 10.1109/EnT52731.2021.00011.

[38] F. Ozkaynak, "A novel random number generator based on fractional order chaotic Chua system", *Elektronika ir Elektrotechnika*, vol. 26, no. 1, pp. 52–57, 2020. DOI: 10.5755/j01.eie.26.1.25310.

[39] Chaotic keys. [Online]. Available: https://eyperoz.github.io/chaotickeys

[40] A. M. Garipcan and E. Erdem, "A gigabit TRNG with novel lightweight post-processing method for cryptographic applications", *The European Physical Journal Plus*, vol. 137, no. 4, art. no. 493, 2022. DOI: 10.1140/epjp/s13360-022-02679-7.

[41] A. M. Garipcan and E. Erdem, "Design, FPGA implementation and statistical analysis of a high-speed and low-area TRNG based on an AES s-box post-processing technique", *ISA Transactions*, vol. 117, pp. 160–171, 2021. DOI: 10.1016/j.isatra.2021.01.054.

[42] E. Tanyildizi and F. Ozkaynak, "A new chaotic s-box generation method using parameter optimization of one dimensional chaotic maps", *IEEE Access*, vol. 7, pp. 117829–117838, 2019. DOI: 10.1109/ACCESS.2019.2936447.