Studies on the Computational Model of PRNG for Data Privacy Risk Mitigation in 5G Networks

Sergiy Gnatyuk[1,2,3], Dinara Ospanova[4], Volodymyr Lytvynenko[5], Zhazira Amirgaliyeva[6] and Nurali Nabot Shohiyon[7]

[1]Yessenov University, microdistrict 32, Main building, Aktau, 130000, Kazakhstan
[2]National Aviation University, Liubomyra Huzara ave.1, Kyiv, 03058, Ukraine[3]Yessenov
[3]State Scientific and Research Institute of Cybersecurity Technologies and Information Protection, Maksyma Zalizniaka str.3/6, Kyiv, 03142, Ukraine
[4]Kazakh Humanitarian Juridical Innovative University, Lenin str, 11, Semey, 070000, Kazakhstan
[5]Kherson National Technical University, Beryslavske shose 24, Kherson, 73008, Ukraine
[6]Al-Farabi Kazakh National University, 71 al-Farabi Ave., Almaty, 050040, Kazakhstan
[7]Dangara State University, Markazi str. 25, Dangara, 735320, Tajikistan

Abstract

Today, pseudo-random number generators (PRNGs) are used in various systems and applications, including as key generators in stream ciphers, blockchain, game industry and others. The implementation of the latest information and communication technology (in particular, modern 5G networks) strengthens the requirements for privacy risk mitigation of critical data and forces the development of new methods and means for cryptographic security. In the paper, a computational model of PRNG was developed and studied. It allows to build efficient algorithms for privacy risk mitigation. Based on this model, software PRNGs have been developed and studied (speed and security parameters were verified). These will be useful for confidentiality ensuring and data privacy risk mitigation in modern 5G networks as well as blockchain technologies.

Keywords

Computational Model, PRNG, Data Privacy, Risk, Algorithm, 5G Networks, Blockchain.

1. Introduction

Today randomness is an important unit in many modern computer applications (especially games, simulations, cryptography). Computers use a form of randomness known as pseudo randomness, it means simulation of randomness. A pseudo random event looks random but is completely predictable or deterministic (result of completely predictable mathematical algorithm). Pseudo-random number generator (PRNG) can be used as key generator in stream ciphers [1] to form large key sequence with small input data of PRNG. This generator creates bit sequence similar to random sequence by statistical parameters. In practice, these sequences are

CITRisk'2021: 2nd International Workshop on Computational & Information Technologies for Risk-Informed Systems, September 16–17, 2021, Kherson, Ukraine
EMAIL: s.gnatyuk@yu.edu.kz (S.Gnatyuk); odm-1778@mail.ru (D.Ospanova); immun56@gmail.com (V.Lytvynenko); zh.amirgaliyeva@gmail.com (Z.Amirgaliyeva); shoev_n@list.ru (N.N.Shohiyon)
ORCID: 0000-0003-4992-0564 (S.Gnatyuk); 0000-0002-2206-7367 (D.Ospanova); 0000-0002-1536-5542 (V.Lytvynenko); 0000-0003-0484-8060 (Z.Amirgaliyeva); 0000-0003-2843-0945 (N.N.Shohiyon)

not random and can be reproduced – key sequence should be as longer as possible and variable. PRNG output must be the function of encryption key and it's very important for privacy ensuring. To generate random numbers by computer the hardware should be used (for example, noise from semiconductor devices; bits of digitized sound from the microphone; intervals between interruptions of external or internal devices; intervals between keystrokes; air temperature on hardware components). New communication standard 5G has improved requirements for confidentiality (privacy) as well as novel secure encryption algorithms and PRNG should be developed (Fig. 1) [2].

Security Elements		WiFi 6 (WPA3)	5G	4G
Terminal identity identification	RADIUS	Terminal - authentication server (AAA)	Terminal - 3GPP AAA	
	Identity identification	Enterprise-level certificate, username/password	Identify authentication information in SIM cards	
Authentication	Security authentication modes	EAP-TLS (certificate) EAP-PEAP (username/password) EAP-AKA (SIM cards on mobile phones)	3GPP and non-3GPP: 5G AKA and EAP-AKA'	3GPP: EPS AKA Non-3GPP: EAP-AKA and EAP-AKA'
Authentication flexibility	Authentication modes in different scenarios	Third-party system authentication such as Portal authentication and WeChat authentication	None	None
NAS security algorithm negotiation	Encryption algorithm	AES	Snow3G, AES, and ZuC	Snow3G, AES, and ZuC
	Algorithm key	256 bits (128 bits for WPA2)	256 bits	128 bits

Figure 1: Privacy ensuring procedures in different network communication standards

2. Related papers analysis

Analysis of related papers in this direction [3-6] shows different approaches in PRNG creation that influences on their characteristics and properties. There are two following categories of PRNGs:

1) Cryptographically secure PRNG based on
- stream ciphers (Dragon-128, SEAL, RC4, RC5, RC6, Grain, Yamb, Phelix);
- block ciphers (AES, ANSI X9.17, DES),
- one way functions (BBS, RSA, Dual_EC_DRBG, GPSSD).

2) *Cryptographically insecure PRNG* based on
- elementary recurents (linear congruental generator, polinomyal congruental generator, additive Fibonacci generator, lagged Fibonacci generator and others);
- operations in finite fields (Galois generator, De Bruijn generator, Golman generator and others).

Most up-to-date cryptographic applications require random numbers, for example key generation, cryptographic nonces, salts in certain signature schemes, including ECDSA, RSASSA-PSS. The security and privacy of 5G is better than in 4G (Fig. 2) [7] by using many secure algorithms.

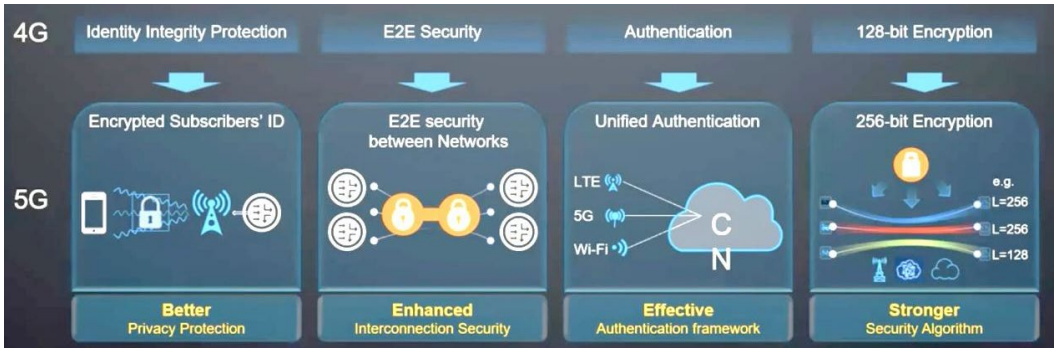

Figure 2: 4G/5G encryption comparison

But there are many relevant tasks related with data privacy ensuring in 4G/5G that can be solved by PRNG development and implementation. From this viewpoint, the main target of this paper is development and study of the computational model of PRNG for data privacy risk mitigation in 5G networks.

3. The main part of the research study

3.1. Theoretical principles of the model construction

On the basis of analysis, as a prototype of computational model well-known PRNG Trivium [8] was chosen. Trivium is a synchronous stream cipher designed by cryptographers C. De Cannière and B. Preneel to provide a flexible trade-off between speed and gate count in hardware, and reasonably efficient software implementation. Trivium was one of the eSTREAM competition winners and its recommended for using in modern communication networks as hardware unit. To improve PRNG Trivium following modifications were proposed:

1. Parameters n, t, e, k were specified and after their fixing new PRNG structure is forming (capacity of operations is changing). All operations are performing not on the bits but on the vectors of some size (bytes).

2. To improve non-linearity parameters substitution operation $S(x)$ was specified. For any new formed PRNG new unique $S(x)$ can be specified.

3. To generate pseudo-random sequences PRNG internal status vector E_i, key vector for sequence generation K and index of the current iteration of forming (generation) i are used.

4. For generation function F_{gen} stage of variables initialization was modified, the dynamic carry shift and substitution operations were specified.

5. For generation function F_{gen} usage of independent functions F_A, F_B, F_C and F_D was proposed, these functions depend on internal status vector of previous generation stage, key vector K and index of current iteration of generation i. Output of the functions F_A, F_B, F_C and F_D will be the data of necessary length (input and output data length will be different). For any new formed PRNG new unique functions F_A, F_B, F_C and F_D can be specified. In fact, these functions are independent byte-oriented PRNGs (without requirements for

cryptographically security), they can work in parallel. For synchronization and optimization of sequence generation procedure the speed of sequence generation should be something like similar.

6. Final stage of sequence m forming and operation ordering were modified as well as the substitution operation was specified.

Various PRNGs can be constructed by modification / fixing parameters n, t, e, k as well all specifying functions F_A, F_B, F_C, F_D and operation $S(x)$.

Computational model description

Let $n, t \in Z_+$, then for generation pseudo-random sequence M, $M \in V_N$, $V_N \in \{0,1\}^N$ with length $N = n \cdot t$ bits it is necessary to form t sequences with length n bit:

$$M = (m_1, m_2, ..., m_{t-1}, m_t),\ m_i \in V_n,\ i = \overline{1,t}.$$

Process of generation every m_i, $m_i \in V_n$, $i = \overline{1,t}$ performs in the following manner:

$$m_i, E_i = F_{gen}(E_{i-1}, K, i),\ i = \overline{1,t},$$

where E_i is internal status vector of PRNG after generation i-th m_i, $E_i \in V_e$, $e \in Z_+$, $E_0 = IV$, IV is initialization vector, $IV \in V_e$, K is key vector for sequence generation, $K \in V_k$, $k \in Z_+$, F_{gen} is function of generation the sequence m_i.

Function $F_{gen}(E, K, i)$ is performing by 2 following stages:

1) variables initialization;
2) sequence forming.

Stage 1 of function $F_{gen}(E, K, i)$ performing. At the beginning the processing of internal status vector is performing E, $E \in V_e$:

$$E = S(E) <<< i,$$

where $x <<< y$ is operation of right dynamic carry shift of argument x on y bits, $S(x)$ is some substitution operation.

Next the internal status vector E of PRNG and key vector K are disintegrated on 4 components:

$$E = (E_a, E_b, E_c, E_d),\ E \in V_e,\ e = a + b + c + d,$$
$$E_a \in V_a,\ E_b \in V_b,\ E_c \in V_c,\ E_d \in V_d,\ a, b, c, d \in Z_+,$$
$$K = (K_a, K_b, K_c, K_d),\ K \in V_k,\ k = a' + b' + c' + d',$$
$$K_a \in V_{a'},\ K_b \in V_{b'},\ K_c \in V_{c'},\ K_d \in V_{d'},\ a', b', c', d' \in Z_+.$$

Vectors E_a, E_b, E_c, E_d and K_a, K_b, K_c, K_d will be used in the next stage of the function $F_{gen}(E, K, i)$.

Stage 2 of function $F_{gen}(E, K, i)$. On this stage the forming of sequence m, $m \in V_n$ is performing. For this objective 4 additional functions $F_A(E_a, K_a, i)$, $F_B(E_b, K_b, i)$,

$F_C\left(E_c, K_c, i\right)$ і $F_D\left(E_d, K_d, i\right)$ are using, where F_A, F_B, F_C and F_D are functions, that have internal status vector and key vector as input data, and sequence with length n bits as output data. These functions can be constructed on the base of non-linear shift registers, block and stream symmetric ciphers, hash functions and others.

In this case, process of sequence m, $m \in V_n$ and new value of internal status vector E, $E \in V_e$ generation in function $F_{gen}\left(E, K, i\right)$ will be follow:

Stage 1. Formation of additional vectors A, B, C і D and calculation of new values of vectors E_a, E_b, E_c, E_d:

$$A, E_a = F_A\left(E_a, K_a, i\right),\ A \in V_n,\ E_a \in V_a,$$
$$B, E_b = F_B\left(E_b, K_b, i\right),\ B \in V_n,\ E_b \in V_b,$$
$$C, E_c = F_C\left(E_c, K_c, i\right),\ C \in V_n,\ E_c \in V_c,$$
$$D, E_d = F_D\left(E_d, K_d, i\right),\ D \in V_n,\ E_d \in V_d.$$

Stage 2. Calculation of the new value internal status vector E, $E \in V_e$:

$$E = \left(E_b, E_d, E_a, E_c\right).$$

Stage 3. Formation of the sequence m, $m \in V_n$:

$$AB = A <<< B,\ AB \in V_n,$$
$$CD = C <<< D,\ CD \in V_n,$$
$$BC = B + CD,\ BC \in V_n,$$
$$AD = AB \oplus D,\ AD \in V_n,$$
$$m = \overline{AD} \oplus S\left(BC\right),\ m \in V_n,$$

where $\oplus$ і $+$ are operations modulo addition 2 and 2^n respectively, $S\left(x\right)$ is substitution operation.

Output of the function $F_{gen}\left(E, K, i\right)$ will be vectors m, $m \in V_n$ and E, $E \in V_e$:

$$\left(m, E\right) = F_{gen}\left(E, K, i\right).$$

In this Section computational model of PRNG was described. Based on PRNG Trivium, this model includes internal status vector and key vector processing, dynamic carry shift and 4 non-linear functions. It allows to construct effective PRNGs for privacy ensuring in modern communication networks (5G networks and others).

3.2. Software PRNG development

On the basis of developed computational model 3 PRNGs *5Gen-1, 5Gen-2* and *5Gen-3* was constructed:

5Gen-1 was constructed with following set of parameters $n = 128$, $a = 128$, $b = 100$, $c = 111$, $d = 173$, $e = a + b + c + d = 512$, $a' = 128$, $b' = 128$, $c' = 128$, $d' = 128$,

$k = a' + b' + c' + d' = 512$. F_A, F_B, F_C and F_D are functions based on the non-linear shift register. As substitution $S(x)$ the operation $S(x) = \left(s_0\left(x_{31}\right), ..., s_0\left(x_0\right)\right)$ was used, where $x_j \in V_{16}$, $j = \overline{0,31}$, S_0 is the substitution on the set V_{16}.

Substitution S_0 is constructed with parameters, presented in Table 1.

Table 1

Parameters for substitution table S_0 construction for *5Gen-1* algorithm

M	*C*	*V*
{ 903, 3206, 640C, C818, 9031, 2063, 40C6, 818C, 319, 632, C64, 18C8, 3190, 6320, C640, 8C81 }	6D71	19CF

5Gen-2 was constructed with following set of parameters $n = 256$, $a = 138$, $b = 120$, $c = 116$, $d = 138$, $e = a + b + c + d = 512$, $a' = 128$, $b' = 128$, $c' = 128$, $d' = 128$, $k = a' + b' + c' + d' = 512$. F_A, F_B, F_C and F_D are functions based on the non-linear shift register. As substitution $S(x)$ the operation $S(x) = \left(s_7\left(x_{63}\right), ..., s_0\left(x_0\right)\right)$ was used, where $x_j \in V_8$, $j = \overline{0,63}$, S_b is substitution on the set V_8, $b = \overline{0,7}$ (by the order 8 different substitution tables are using).

Substitution S_i, $i = \overline{0,7}$ is constructed with parameters, presented in Tables 2-9.

Table 2

Parameters for substitution table S_0 construction for *5Gen-2* algorithm

M	*C*	*V*
{ 29, 52, A4, 49, 92, 25, 4a, 94 }	07	D8

Table 3

Parameters for substitution table S_1 construction for *5Gen-2* algorithm

M	*C*	*V*
{ 70, E0, C1, 83, 7, E, 1C, 38 }	A2	44

Table 4

Parameters for substitution table S_2 construction for *5Gen-2* algorithm

M	*C*	*V*
{ 3E, 7C, F8, F1, E3, C7, 8F, 1F }	72	63

Table 5
Parameters for substitution table S_3 construction for *5Gen-2* algorithm

M	C	V
{ E3, C7, 8F, 1F, 3E, 7C, F8, F1 }	43	9B

Table 6
Parameters for substitution table S_4 construction for *5Gen-2* algorithm

M	C	V
{ E5, CB, 97, 2F, 5E, BC, 79, F2 }	A0	8C

Table 7
Parameters for substitution table S_5 construction for *5Gen-2* algorithm

M	C	V
{ AB, 57, AE, 5D, BA, 75, EA, D5 }	7B	C6

Table 8
Parameters for substitution table S_6 construction for *5Gen-2* algorithm

M	C	V
{ 91, 23, 46, 8C, 19, 32, 64, C8 }	ED	B0

Table 9
Parameters for substitution table S_7 construction for *5Gen-2* algorithm

M	C	V
{ F8, F1, E3, C7, 8F, 1F, 3E, 7C}	18	75

5Gen-3 was constructed with following set of parameters $n = 128$, $a = 128$, $b = 128$, $c = 128$, $d = 128$, $e = a + b + c + d = 512$, $a' = 128$, $b' = 128$, $c' = 128$, $d' = 128$, $k = a' + b' + c' + d' = 512$. F_A, F_B, F_C and F_D are functions based on AES-128 algorithm. As substitution $S(x)$ the operation $S(x) = \left(s_0\left(x_{63}\right), \ldots, s_0\left(x_0\right)\right)$ was used, where $x_j \in V_8$, $j = \overline{0,63}$, S_0 – підстановка на множині V_8.

Substitution S_0 is constructed with parameters, presented in Tables 10.

Table 10
Parameters for substitution table S_0 construction for *5Gen-3* algorithm

M	C	V
{ 20, 40, 80, 1, 2, 4, 8, 10}	59	6B

3.3. Experimental study and discussion

Statistical parameters investigation

Statistical parameters of developed PRNGs were investigated using traditional techniques NIST STS [9] and DIEHARD [13]. Given results were compared with testing results of generator BBS as well as stream ciphers SNOW [10] and Trivium (used in 5G networks). NIST STS are used to determine the qualitative and quantitative features of the sequences randomness. Three basic criteria are used to draw conclusions about passing random sequences of statistical tests are following:

- Criterion for decision-making based on the establishment of some threshold level.
- Criterion based on establishing a fixed confidence interval.
- Criterion for some appropriate statistical test probability value P-value.

The statistical test is based on the verification of null hypothesis H_0 – that the sequence under study is random. Alternative hypothesis H_1 is also provided – the sequence under study is not random. Therefore, the generated sequence is examined by a set of tests, each of which concludes whether the hypothesis H_0 is rejected or accepted. For each test, adequate randomness statistics are selected based on which the hypothesis H_0 is further rejected or accepted. Theoretically, the distribution of statistics for the null hypothesis is calculated using mathematical methods. The critical value is then determined from such an exemplary distribution. When performing the test, the value of the test statistic is calculated, which is compared to the critical value. In case of exceeding the test critical value over the reference, hypothesis H_1 is accepted, otherwise – hypothesis H_0.

For the experiments, the following input parameters were selected for the use of NIST STS:

1. The length of the test sequence n=10^6 bit.
2. The number of sequences being tested m=100.
3. Significance level $\alpha = 0,01$.
4. Number of tests q=188, among them: Frequency − 1, Block Frequency − 1, Cumulative Sums − 2, Runs − 1, Longest Run − 1, Rank − 1, FFT − 1, Non Overlapping Template − 148, Overlapping Template − 1, Universal − 1, Approximate Entropy − 1, Random Excursions − 8, Random Excursions Variant − 18, Serial − 2, Linear Complexity − 1.

The confidence interval rule was applied, the lower bound being 0.96015. For every algorithms 10 files with sequences (100 Mbit) and investigated by NIST STS technique.

The results of testing are presented on Fig. 3-5.

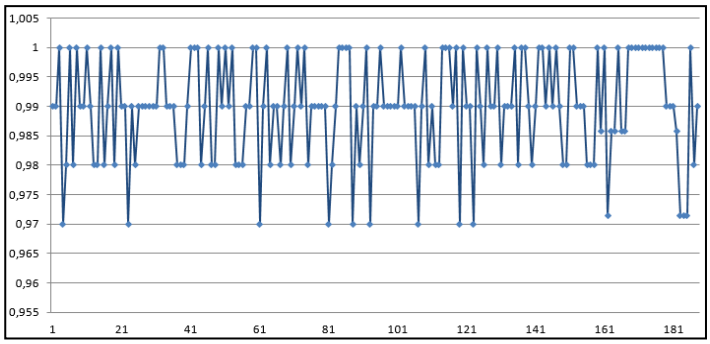

Figure 3: Statistical portrait of *5Gen-1* algorithm by NIST STS technique

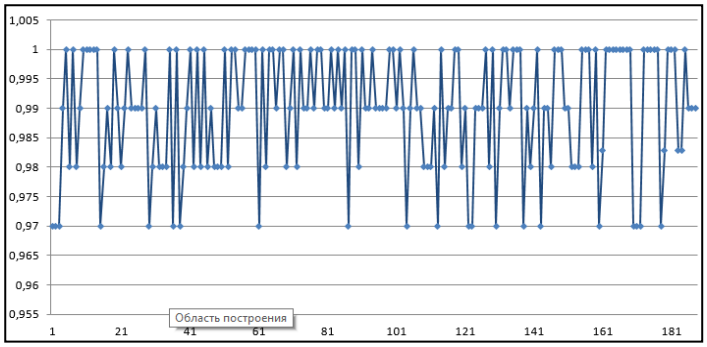

Figure 4: Statistical portrait of *5Gen-2* algorithm by NIST STS technique

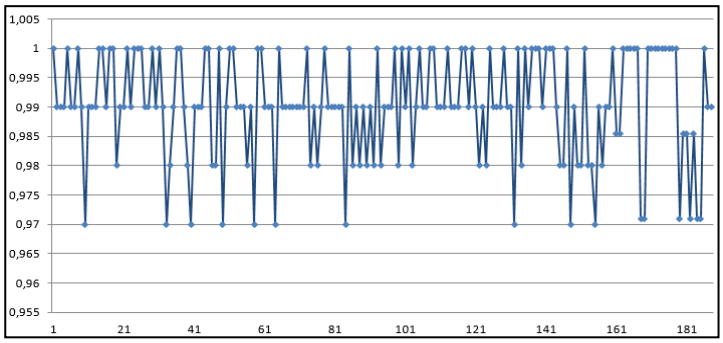

Figure 5: Statistical portrait of *5Gen-3* algorithm by NIST STS technique

Also developed algorithms were investigated by DIEHARD technique (one of the best practice and traditional approach in cryptography for measuring quality of PRNG and statistical security). Results of testing for *5Gen-3* algorithms presented on Fig. 6 (other two algorithms *5Gen-1* and *5Gen-2* showed 100% results).

Test name	ntup	P - value	Assessment
Birthdays	0	0.46632136	Passed
OPERM5	0	0.67512015	Passed
32x32 Binary Rank	0	0.69527276	Passed
6x8 Binary Rank	0	0.10123586	Passed
Bitstream	0	0.98654155	Passed
OPSO	0	0.79734863	Passed
OQSO	0	0.9987444	Weak
DNA	0	0.89704567	Passed
Count the 1s (stream)	0	0.63873474	Passed
Count the 1s (byte)	0	0.94012168	Passed
Parking Lot	0	0.24635381	Passed
Minimum Distance (2d Circle)	2	0.89675937	Passed
Minimum Distance (3d Sphere)	3	0.37439327	Passed
Squeeze	0	0.78310853	Passed
Sums	0	0.81065417	Passed
Runs	0	0.99245722	Passed
Runs	0	0.45528189	Passed
Craps	0	0.75941456	Passed
Craps	0	0.95427721	Passed

Figure 6: Statistical testing results of *5Gen-3* algorithm by DIEHARD technique

Results presented on Fig. 3-5 as well as detail experimental data (Table 11) show that developed algorithms *5Gen-1, 5Gen-2* and *5Gen-3* have passed complex control by NIST STS technique and show better results (in some cases) than existed and well-known algorithms [11-12].

Results presented on Fig. 6 show that developed algorithms have passed complex security control by DIEHARD technique as well as verified good quality of PRNG and statistical security of potential systems based on these algorithms [14-16].

Table 11
Results of statistical parameters investigation by NIST STS technique

PRNG	Test total amount, where passed	
	99% sequences	96% sequences
BBS	133.4 (70.96%)	188 (100%)
SNOW	134.8 (71.70%)	188 (100%)
Trivium	130.1 (69.20%)	187.6 (99.78%)
5Gen-1	134.1 (71,32%)	188 (100%)
5Gen-2	136.4 (72,55%)	187.8 (99.89%)
5Gen-3	137.3 (73,03%)	188 (100%)

Speed parameters investigation

For investigation developed PRNGs 5Gen-1, 5Gen-2, 5Gen-3 were realized as software tools using programming language C++. Given results were compared with results of well-known SNOW generator. Experimental study was carried out using work station Intel Core i3-3220 3.3GHz and files with different size. Average results are presented on the Table 12.

Table 12
Results of speed parameters investigation

PRNG	File 1, 1 MB		File 2, 10 MB		File, 100 MB	
	t ,s	v , MB/s	t ,s	v , MB/s	t ,s	v , MB/s
SNOW	0.011	90.91	0.107	93.46	1.01	99.01
5Gen-1	0.009	111.11	0.091	109.89	0.88	113.64
5Gen-2	0.010	100.00	0.098	102.04	0.92	108.70
5Gen-3	0.014	71.43	0.112	89.29	0.99	101.01

As we can see from Table 12 speed of the developed algorithms are higher in comparison with SNOW till 21% outside two results of 5Gen-3 algorithm.

4. Conclusions

In this paper analysis of different approaches in PRNG creation and implementation was carried out. Two categories of PRNGs were defined (cryptographically secure and cryptographically insecure) as well cryptographic applications that require random numbers were defined (key generation, cryptographic nonces, salts in certain signature schemes). But there are many

relevant tasks related with data privacy risk mitigation and confidentiality ensuring in 4G/5G that can be solved by advanced PRNG development and implementation.

Computational model of PRNG was developed. Based on PRNG Trivium, this model includes internal status vector and key vector processing, dynamic carry shift and 4 non-linear functions. It allows to construct effective PRNGs for data privacy risk mitigation in modern communication networks (5G networks and others).

On the basis of developed computational model 3 PRNGs *5Gen-1, 5Gen-2* and *5Gen-3* was constructed and realized as software tools. These PRNGs have passed complex statistical testing by NIST STS technique as well as speed parameters investigation (developed algorithms are higher in comparison with SNOW till 21%).

Additionally, developed algorithms were investigated by DIEHARD technique (one of the best practice and traditional approach in cryptography for measuring quality of PRNG and statistical security). These algorithms have passed complex security control by DIEHARD technique as well as verified good quality of PRNG and statistical security of potential systems based on these algorithms.

References

[1] Z.Hu, S.Gnatyuk, T.Okhrimenko, S.Tynymbayev, M.Iavich, High-speed and secure PRNG for cryptographic applications, International Journal of Computer Network and Information Security, Volume 12, Issue 3, 2020, pp. 1-10

[2] Security Comparison Between Wi-Fi 6 and 5G, https://forum.huawei.com/enterprise/en/security-comparison-between-wi-fi-6-and-5g/thread/615836-869

[3] S.Yevseyev, R.Koroliov, M.Krasnyanska, Analysis of current methods of generating pseudorandom sequences, Eastern-European Journal of Advanced Technology, 3/4 (45), 2010, pp. 11-15

[4] W.Z.Yang, Z.J.Zheng, PRNG based on the variant logic" 7th International Conference on Communications and Networking in China, 2012, pp. 202-205, doi: 10.1109/ChinaCom.2012.6417476

[5] R.Hobincu, O.Datcu, C.Macovei, Entropy global control for a chaos based PRNG, 42nd International Conference on Telecommunications and Signal Processing (TSP), 2019, pp. 432-435, doi: 10.1109/TSP.2019.8768818

[6] Y.S.S.Risqi, S.Windarta, tatistical test on lightweight block cipher-based PRNG, 11th International Conference on Telecommunication Systems Services and Applications (TSSA), 2017, pp. 1-4, doi: 10.1109/TSSA.2017.8272925

[7] 5G security is fundamentally better than LTE security but the attack surface is massively increased, https://www.rcrwireless.com/20200416/5g/huawei-cto-5g-securiuty-standards

[8] C.De Cannière, B.Preneel, TRIVIUM – Specifications, eSTREAM, ECRYPT Stream Cipher Project, Report 2005/030 (2005), http://www.ecrypt.eu.org/stream

[9] NIST STS, Download documentation and software https://github.com/kravietz/nist-sts

[10] P. Ekdahl, T. Johansson, SNOW. A new stream cipher, Proceedings of the First NESSIE Workshop, NESSIE, 2000

[11] S.Gnatyuk, T.Okhrimenko, A.Fesenko et al, Experimental Study of Secure PRNG for Q-trits Quantum Cryptography Protocols, Proceedings of the 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT 2020), Kyiv, Ukraine, May 14, 2020, pp. 183-188

[12] J.M.Mcginthy, A.J.Michaels, Further Analysis of PRNG-Based Key Derivation Functions, IEEE Access, vol. 7, 2019, pp. 95978-95986
[13] P.S.Paul, M.Sadia, M.S.Hasan, Design of a Dynamic Parameter-Controlled Chaotic-PRNG in a 65 nm CMOS process, IEEE 14th Dallas Circuits and Systems Conference (DCAS), 2020, pp. 1-4, doi: 10.1109/DCAS51144.2020.9330647
[14] Z.Hua, Y.Zhou, One-Dimensional Nonlinear Model for Producing Chaos, in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 65, no. 1, 2018, pp. 235-246, Jan. doi: 10.1109/TCSI.2017.2717943
[15] M.Iavich, A.Gagnidze, G.Iashvili, S.Gnatyuk, V.Vialkova, Lattice based Merkle, CEUR Workshop Proceedings, vol. 2470, 2019, pp. 13-16
[16] S.Ergün, Security analysis of a random number generator based on a double-scroll chaotic circuit", 16th International Symposium on Communications and Information Technologies (ISCIT), 2016, pp. 123-126, doi: 10.1109/ISCIT.2016.7751605