BARDOSHLI KRIPTOGRAFIK KALITLARNI YARATISHDA PSEVDO-TASODIFIY RAQAMLAR GENERATORLARIDAN FOYDALANISH

[1]Bozorov A.X., [2]Muxamadiyev F.R.
[1,2]O'zbekiston milliy universiteti, tayanch doktarant

https://doi.org/10.5281/zenodo.7857589

Abstract. *Random numbers have long been used in many fields until today. This includes scientific research and computer modeling of physical processes, numerical analysis, and even computer games that require the ability to generate random samples.*

Keywords: *random numbers, scientific research, computer modeling, phisical processes, numerical analysis.*

Tasodifiy raqamlar bugungi kungacha ko'plab sohalarda uzoq vaqtdan beri qo'llanilgan. Bunga tasodifiy namunalarni shakllantirish qobiliyatiga muhtoj bo'lgan ilmiy tadqiqotlar va jismoniy jarayonlarni kompyuter modellashtirish, raqamli tahlil va hatto kompyuter o'yinlari ham kiradi. Xuddi shu tasodifiy sonlarni olish uchun psevdo-tasodifiy sonlar generatorlari (PRNG) foydalanamiz. PRNG Generatorlari quyidagi talablarga javob berishi kerak:

- Ketma-ketlik hal qilinayotgan muammo ichida aylanmasligini ta'minlash uchun etarlicha uzoq muddat.
- Davrning uzunligi matematik jihatdan isbotlanishi kerak.
- Samaradorlik - algoritm tezligi va past xotira xarajatlari.
- Reproduktivlik - oldindan yaratilgan raqamlar ketma-ketligini istalgan sonda qayta ishlab chiqarish qobiliyati.
- Portativlik - bu turli apparat va operatsion tizimlarda bir xil operatsiya.
- Har qanday i qiymati uchun X_n elementini belgilashda ketma-ketlikning X_{n+1} elementini olish tezligi (ketma-ketlikni bir nechta oqimlarga bo'lish uchun)

Ushbu talablarga javob beradigan juda ko'p miqdordagi PRNG mavjud bo'lsa-da, ularning faqat kichik bir qismi kriptografik jihatdan kuchli, ya'ni tasodifiylik va maxsus talablar bo'yicha statistik testlar asosida taxlil qilinadi. [3] Tasodifiy sonlar generatorlarini o'rganishda ya'ni raqamlarni olish usuliga ko'ra ularni quyidagi turlarga bo'lamiz:

1. Uskuna
2. Jadval
3. Algoritmik

1. Jadval generatorlari tasodifiy sonlar manbai sifatida tasdiqlangan o'zaro bog'liq bo'lmagan raqamlarni o'z ichiga olgan oldindan tayyorlangan jadvallardan foydalanadi.

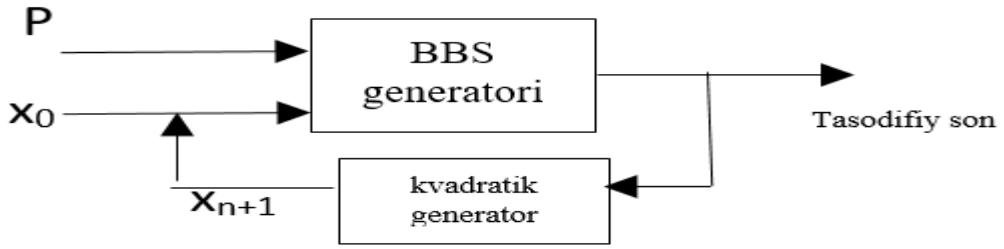
2. Haqiqiy tasodifiy ketma-ketliklarning apparat (uskuna) generatorlari - tasodifiy sonlarni yaratish uchun ba'zi jismoniy jarayonlarning parametrlarini o'lchashdan foydalanadigan qurilma.

3. Tasodifiy sonlar generatorlarining apparat vositalarining qimmatligi tufayli ko'p hollarda entropiya manbai sifatida algoritmik alternativ bo'lgan generatorlar - psevdotasodifiy sonlar generatoridan foydalanadi. Ularni biz psevdotasodfiy ketma-ketlikni shakllantiruvchi generatorlar ma'lum algoritmik tipga ega.

Hozirgi kundagi yuqori samaradorlikka ega shifrlash algoritmlarida shakllantirish asosida yuqorida keltirilgan usullar yotadi. Jumladan, A5 algoritmi, Hughes XPD/KPD algoritmi, Fish algoritmi, Pike algoritmi, Mush algoritmi, katakli avtomat asosidagi PTKG, Vixr Mersenna usuli,

FIPS–186 algoritmi, ANSI X9.17 algoritmi, BBS algoritmi, RSA algoritmi, Yarrow–160 algoritmi va h.k. algoritmlarini keltirish mumkin. Yuqorida keltirilgan algoritm va usullardan kelib chiqqan holda shakllantirish uchun samarali va qulay bo'lgan usulni tanlab olish kerak. Bu borada bir martali parollarni shakllantirish uchun psevdotasodfiy ketma-ketlik generatori BBS algoritmi olindi. [3]

Hozirgi vaqtda eng sodda va eng samarali kriptografik jihatdan kuchli psevdo — tasodifiy sonlar generatori BBS generatoridir (Blum — Blum-Shub), yaratuvchilar nomi bilan atalgan L. Blum, M. Blum va M. Shuba va kvadratik ajratmalar nazariyasiga asoslangan modulli algoritm.[4]



1-rasm. BBS generatori tasvirlangan

BBS generatorini quyidagi formula asosida hisoblashimiz mumkin:

$$x_i = x_{i-1}^2 \bmod n \qquad (1)$$

Keyinchalik, iterativ formula bilan berilgan raqamlari hisoblanadi. Kerakli l uzunlikdagi P psevdo tasodifiy ketma-ketlikning bitlari sifatida x raqamlarning kichik bitlari ishlatiladi.[5]

$P = b_0 b_1 b_2 ... b_{l-1}$. - ketmat-ketlikdan quydagi formulani hosil qilamiz:

$$b_i = x_i \bmod 2, \quad i = 0,1,...,l-1. \qquad (2)$$

Bu yerda $n = p * q$ va tub sonlar p, $q = 3 \bmod 4$

p	X_i	B_i	p	X_i	B_i
1	20749	1	11	137922	0
2	143135	1	12	123175	1
3	177671	0	13	8630	0
4	97048	0	14	114386	0
5	89992	1	15	14863	1
6	86649	1	16	133015	1
7	45663	1	17	106065	1
8	69442	0	18	45870	0
9	186894	0	19	137171	1
10	177046	0	20	48060	0

C++ dasturlash tilidagi quyidagi manba matnda BBS generatorini amalga oshirish ko'rsatilgan. Dastur ikkita tasodifiy sonni hisoblab chiqadi va ularni konsolda ko'rsatadi. U P ketma-ketlikning istalgan bitini samarali bevosita aniqlash imkonini beradi. Har qanday x-ni faqat x0 ning boshlang'ich qiymati va i seriya raqami asosida hisoblash mumkin:

$$x_i = x_0^{2^i \bmod (p-1)(q-1)} \bmod n. \quad (3)$$

Agar butun sonlarni faktorizatsiya qilish qiyin bo'lsa (siz kutganingizdek), u holda katta *M* bo'lgan BBSlar oqilona miqdordagi hisoblash orqali topilishi mumkin bo'lgan tasodifiy bo'lmagan naqshlardan xoli natijaga ega bo'ladi. BBS algoritmi quydagi afzaliklarga ega:

- shifrlash uchun juda sekin, kalitlarni generatsiya qilish uchun yaxshi
- taxmin qilish mumkin emas, keyingi bit sinovdan o'tkaziladi (oldingi k bitga asoslangan holda taxmin qiling).
- n xavfsizlik faktorining murakkabligiga bog'liq
- har qanday bit to'plamini hisobga olsak, oldindan aytib bo'lmaydi
- sekin, chunki juda katta raqamlardan foydalanish kerak.

Dastlabki parametrlarni to'g'ri tanlash bilan BBS algoritmi psevdo tasodifiy ketma-ketliklar uchun barcha statistik mezonlarga javob beradi. BBS generatorining xavfsizligi faktorizatsiya muammosiga asoslanadi. BBSni buzishi mumkin bo'lgan harakatlar bo'lishi mumkin, ammo bu haraktlar deyarli imkonsiz deb hisoblanadi.

Demak, shunday xulosaga ega bo’lamizki, an'anaviy psevdotasodifiy sonlar generatorlarini kriptografik jihatdan xavfsiz psevdotasodifiy generatorlar sifatida ishlatish mumkin emas, chunki ular keyingi yaratilgan bitning oldingisidan mustaqil bo'lish xususiyatiga ega emas va teskari muhandislik usullari bilan ishonchsizdir. Teskari muhandislik (yoki teskari muhandislik) - bu qurilma yoki dasturni, shuningdek, uning qanday ishlashini tushunish va ko'pincha qurilma, dastur yoki shunga o'xshash funktsiyalarga ega boshqa ob'ektni nusxa ko'chirmasdan qayta ishlab chiqarish uchun uning hujjatlarini o'rganishni talab etadi.

REFERENCES

1. Lenore Blum, Manuel Blum, und Michael Shub: A Simple Unpredictable Pseudo-Random Number Generator, SIAM Journal on Computing, Band 15, Nr. 2, Seiten 364–383, Mai 1986.
2. Lenore Blum, Manuel Blum, und Michael Shub: Comparison of two pseudo-random number generators, Advances in Cryptology: Proceedings of Crypto ’82.
3. https\\:habr.com\Обзорная экскурсия в криптографически стойкие генераторы псевдослучайных чисел – С. 1-5.
4. https://studme.org/239571/informatika/upravlenie_kriptograficheskimi_klyuch C. 1-4.
5. https://www.slideserve.com/bonnie/stream-cipher-diagram – С. 1-3.