Perbandingan Cryptographic Random Number Generator Vs True Random Number Generator

AppZaza & SecureRandom Vs Random.org & Quantumrandom

Ahmad Faishol Huda
13516094
Studi Teknik Informatika
Institut Teknologi Bandung
Bandung, Indonesia
13516073@std.stei.itb.ac.id

***Abstract*—Random Number Generator adalah sebuah algoritma untuk mendapatkan sebuah nilai secara random/acak. Namun *Random Number Generator* sebenarnya tidak benar-benar menghasilkan nilai secara *random*. Di dunia modern *Random Number Generator* diperoleh dari sebuah fungsi deterministik yang memiliki pola sehingga hasilnya bisa diprediksi. Sedangkan True Random Number Generator sendiri seharusnya tidak menggunakan fungsi tersebut. True Random Number Generator seharusnya memanfaatkan sebuah alat yang menangkap entropi dari fenomena alam seperti bunyi atmosfer dan bunyi termal. *Cryptographic Random Number Generator* adalah algoritma pseudo random generator yang didesain untuk mendapatkan nilai yang bisa dipakai dalam kriptografi dengan secure. Nilai ini biasanya dihasilkan dari entropi yang didapatkan dari sumber entropi yang berkualitas, biasanya API Random dari sistem. Untuk mengukur algoritma mana yang lebih baik akan dilakukan perbandingan dan test dengan NIST Test Suite untuk Random Number Generator.**

Keywords— Random Number Generator, True Random Number Generator, Cryptographic Random Number Generator, NIST Test Suite.

I. Pendahuluan

Random Number Generator adalah sebuah algoritma untuk mendapatkan sebuah nilai secara random/acak. Namun *Random Number Generator* sebenarnya tidak benar-benar menghasilkan nilai secara *random*. Di dunia modern *Random Number Generator* diperoleh dari sebuah fungsi deterministik yang memiliki pola sehingga hasilnya bisa diprediksi. Sedangkan *True Random Number Generator* sendiri seharusnya tidak menggunakan fungsi tersebut. T*rue Random Number Generator* seharusnya memanfaatkan sebuah alat yang menangkap entropi dari fenomena alam seperti bunyi atmosfer dan bunyi termal.

Sekarang sudah ada beberapa service *Random Number Generator* yang mengaku bahwa mereka adalah *True Random Number Generator* dengan memanfaatkan bunyi atmosfer, fluktuasi vakum, dan lain lain.

II. Dasar Teori

A. Random Number Generator

Random Number Generator dalam dunia informatika adalah sebuah algoritma untuk menghasilkan sebuah nilai secara random. *Random Number Generator* dibagi menjadi 2 macam yaitu, *Pseudo Random Number Generator* dan *True Random Number Generator.*

Pseudo Random Number Generator adalah sebuah algoritma *Random Number Generator* yang cara membangkitkan nilai randomnya menggunakan sebuah fungsi deterministik dengan sebuah seed. Karena menggunakan fungsi deterministik hasil dari algoritma ini tidak benar benar random dan bisa diprediksi.

True Random Number Generator adalah *Random Number Generator* yang cara membangkitkan nilai randomnya menggunakan bunyi/*noise* dari fenomena alam. Beberapa macam bunyi/*noise* yang bisa dan sudah digunakan untuk algoritma ini adalah bunyian atmosfer, vakum, elektromagnetik dan fenomena kuantum lainnya. Dengan memanfaatkan bunyian ini nantinya kita akan mendapatkan seed yang sangat acak sehingga menghasilkan nilai yang acak juga.

Cryptographic Random Number Generator adalah sebuah algoritma *pseudo random generator* yang didesain khusus untuk digunakan dalam bidang kriptografi. Sudah jelas bahwa dalam kriptografi kadang dibutuhkan nilai random yang berkualitas, near-true random, untuk itu dibuatlah algoritma ini. Algoritma ini biasanya memanfaatkan seed dari sebuah entropi,

biasanya entropi yang dipakai adalah API random dari sistem.

B. quantumrandom

quantumrandom adalah sebuah project(module) pada python. Project ini dalam pembangkitan nilai randomnya tersambung dengan website http://qrng.anu.edu.au. Pembangkitan random disini dilakukan dengan menghitung fluktuasi kuantum pada vakum.

C. SecureRandom(Java)

Securerandom di Java adalah sebuah kelas yang mengimplementasikan sebuah pembangkit angka acak yang kuat untuk kebutuhan kriptografi. Angka acak kriptografis yang kuat minimal sesuai dengan uji generator angka acak secara statistik statistik yang ditentukan dalam FIPS 140-2, Persyaratan Keamanan untuk Modul Kriptografi, bagian 4.9.1.

Selain itu, SecureRandom harus menghasilkan keluaran non-deterministik. Oleh karena itu setiap materi unggulan yang diteruskan ke objek SecureRandom harus tidak dapat diprediksi, dan semua urutan output SecureRandom harus kuat secara kriptografis.

D. Random.org

Random.org adalah sebuah service yang menyediakan *random number generator*. Pembangkitan nilai random pada service ini adalah dengan memanfaatkan noise/suara dari atmosfer .

E. AppZaza

AppZaza adalah sebuah service yang menyediakan random number generator. Pembangkitan nilai di sini dilakukan secara kriptografik.

F. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications (NIST)

NIST menyediakan sebuah Test Suite untuk melakukan test terhadap Random Number Generator untuk mengukur keamanan sebuah random number generator untuk digunakan dalam kriptografi.

NIST memberikan suatu kumpulan metode untuk melakukan pengujian pembangkit nilai acak. Beberapa pengujian yang dilakukan berupa pengujian pada level bit. Pengujian yang dilakukan terdapat 15 metode uji [3], yaitu

1. Pengujian Frekuensi (Monobit)
2. Pengujian Frekuensi dengan sebuah Blok
3. Pengujian Eksekusi (Runs Test)
4. Pengujian untuk Longest-Run-of-One's dalam sebuah Blok
5. Pengujian Rank Matriks Binary (Binary Matrix Rank)
6. Pengujian Discrete Fourier Transform (Spectral)
7. Pengujian Pencocokan Non-overlapping Template
8. Pengujian Pencocokan Overlapping Template
9. Pengujian “Universal Statistical” Maurer
10. Pengujian Kompleksitas Linear
11. Pengujian Serial
12. Pengujian Aproksimasi Entropi
13. Pengujian Jumlah Kumulatif (Cumulative Sums)
14. Pengujian Ekskursi Acak
15. Pengujian Varian Ekskursi Acak

III. Perbandingan dan Analisis

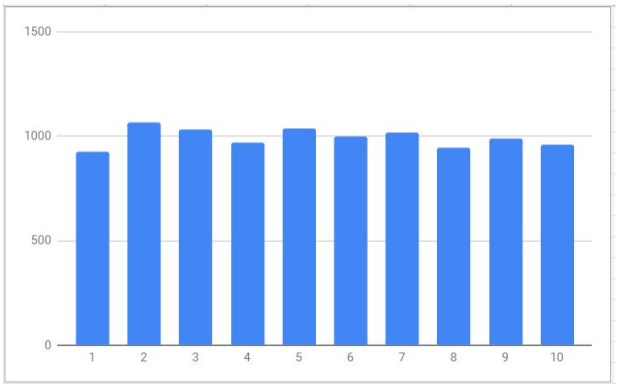
Pada Pengujian ini akan dilakukan dengan menggunakan test suite NIST. Tes yang akan dilakukan ada 3 test. yaitu Tes pengujian frekuensi,

A. Pengujian Frekuensi dengan sebuah Blok

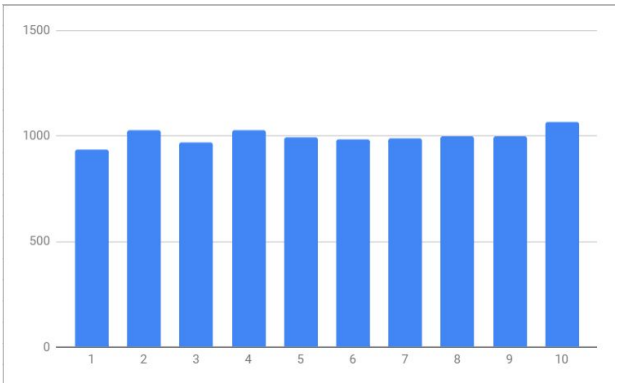
Pada pengujian berdasarkan frekuensi akan dilakukan dengan :

1. Pengujian dengan range 1-10 sebanyak 10000x

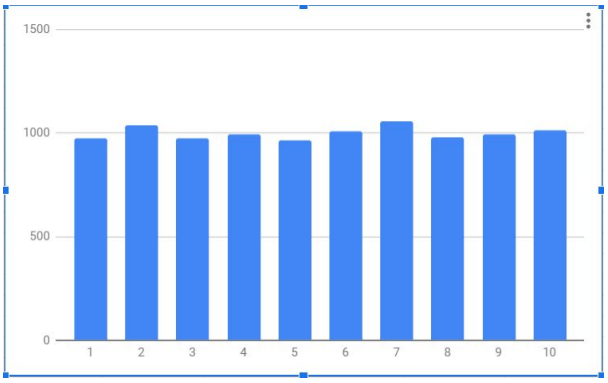
➢ quantumrandom



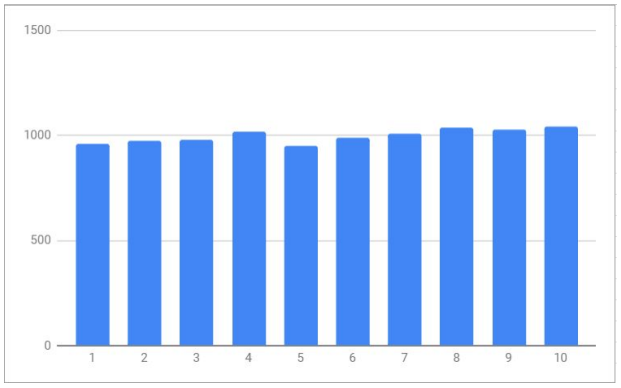
➢ SecureRandom



➢ random.org



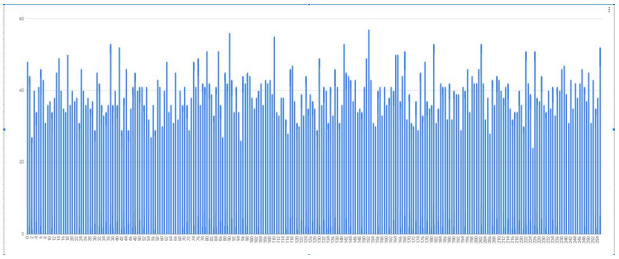
➢ ApZaza



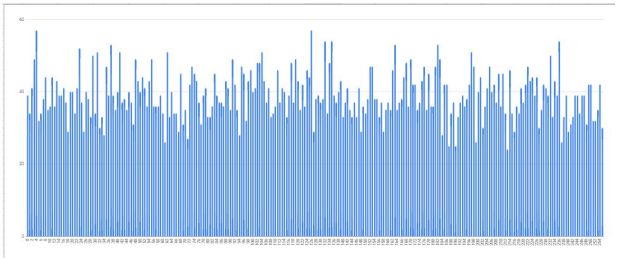
Pada gambar grafik diatas terlihat bahwa masing masing algoritma/services menghasilkan grafik yang cukup bagus. Pada grafik-grafik diatas persebaran nilai masih cukup merata dan tidak ada nilai yang berbeda jauh. Sehingga sulit diambil kesimpulan melalui test ini.

2. Pengujian dengan range 0-255 sebanyak 10000x

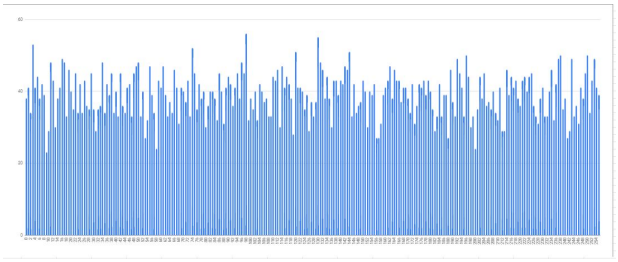
➢ quantumrandom



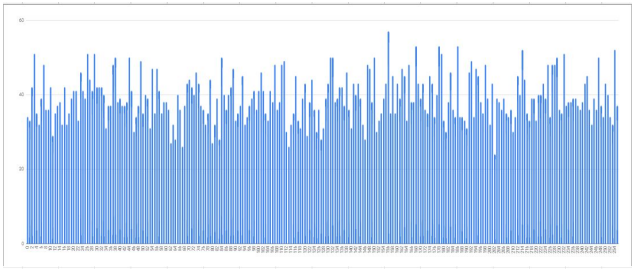
➢ SecureRandom



➢ random.org



➢ ApZaza



Pada gambar grafik diatas terlihat bahwa masing masing algoritma/services menghasilkan grafik masih cukup bagus. Pada grafik-grafik meskipun terdapat beberapa nilai yang cukup berbeda dengan nilai lain, dan range nya juga cukup tinggi namun jumlahnya masih tidak terlalu jauh dengan rata rata nya.

Meskipun terdapat perbedaan range dan juga deviasi pada masing masing grafik, nilai range dan juga deviasinya masih cukup dekat sehingga tidak dapat diambil kesimpulan yang cukup meyakinkan.

B. Pengujian frekuensi MonoBit

Pengujian frekuensi bit biasanya dilakukan untuk menghitung P value. Pada tes ini akan dilakukan perbandingan antara quantumrandom dengan SecureRandom. quantum random akan mewakili True Random Number Generator dan Java Secure Random akan mewakili Cryptographic Random Number Generator.

P-values dianggap gagal jika nilainya dibawah 0.01. Nilai P-values yang kecil berarti nilai deviasi besar dari proporsi nilai 1 dan 0 dalam satu blok.

Percobaan ke-	P-values			
	quantumrandom	SecureRandom	Random.org	AppZaza
1	0.99998	0.045500	0.654641	0.597941
2	0.35027	0.31731	0.695616	0.338916

3	0.79691	0.31731	0.451577	0.394877
4	0.89371	1.00000	0.548377	0.491677
5	0.32849	0.04550	0.528497	0.471797
6	0.67492	0.31731	0.329584	0.272884
7	0.91063	1.00000	0.914631	0.857931
8	0.78796	0.04550	0.442627	0.385927
9	0.39772	1.00000	0.263189	0.206489
10	0.28185	0.04550	0.816385	0.873085
Rata	0.64224	0.04550	0.564512	0.489152

Dari data diatas terlihat bahwa masing masing generator memiliki P-value yang selalu diatas 0.01, sehingga lolos dari test NIST ini yaitu nilai P-values >0.01. Sedangkan untuk perbandingan dari keduanya terlihat bahwa quantum random lebih sering mendapatkan nilai diatas 0.75 dan nilai rata-rata p-values nya lebih tinggi daripada nilai rata rata p-values dari SecureRandom. Niali p-values pada SecureRandom lebih rendah meskipun pernah mencapai nilai 1. Ini menunjukkan bahwa quantumrandom yang menggunakan bunyi fluktuasi vakum untuk mengaplikasikan True Random Number Generator dapat menghasilkan bit-bit acak secara seimbang lebih baik daripada SecureRandom yang mengaplikasikan Cryptographic Random Number Generator.

Nilai rata-rata P-values pada Random.org juga lebih tinggi daripada nilai milik AppZaza, yaitu 0.564512 dibanding 0.489152, meskipun tidak terlalu berbeda jauh ini tetap menunjukkan bahwa Random.org yang memanfaatkan suara/noise dari Atmosfer untuk mendapatkan nilai randomnya jauh lebih baik daripada AppZaza yang menggunakan Cryptographic Random Generator.

C. Perbandingan run test

Percobaan ke-	quantumrandom	SecureRandom
1	0.965692	1.00000
2	0.644020	0.00000
3	0.493235	0.02868
4	0.644020	0.14402
5	0.151626	0.751626
6	0.713173	0.213173
7	0.942580	0.44258
8	0.493235	0.343235
9	0.401430	0.151432
10	0.849220	0.349222
rata	0.629823	0.342396

Dari data diatas terlihat bahwa masing masing generator memiliki P-value yang selalu diatas 0.01, sehingga lolos dari test NIST ini yaitu nilai P-values >0.01. Sedangkan untuk perbandingan dari keduanya terlihat bahwa quantum random lebih sering mendapatkan nilai diatas 0.5 dan nilai rata-rata p-values nya lebih tinggi daripada nilai rata rata p-values dari SecureRandom. Niali p-values pada SecureRandom lebih rendah meskipun pernah mencapai nilai 1. Ini menunjukkan bahwa quantumrandom yang menggunakan bunyi fluktuasi vakum untuk mengaplikasikan True Random Number Generator dapat menghasilkan bit-bit acak secara seimbang lebih baik daripada SecureRandom yang mengaplikasikan Cryptographic Random Number Generator.

IV. Kesimpulan dan Saran

Secara teori, True Random Number Generator pasti lebih baik performanya daripada Cryptographic Random Number Generator. Bagaimanapun juga Cryptographic Random Number Generator adalah sebuah Pseudo Random Number Generator dimana menggunakan seed untuk membangkitkan nilai randomnya meskipun sudah didesain untuk digunakan dalam bidang kriptografi dan dianggap secure.

Dari percobaan, perbandingan, dan juga analisis sebelumnya juga terlihat bahwa performa dari True Random Generator lebih baik daripada Cryptographic Random Generator terutama pada *test bit dan Test Run.*

Pada Test monobit P-values quantumrandom lebih stabil dan lebih tinggi daripada p-value dari SecureRandom, juga P-Values Random.org lebih tinggi daripada AppZaza..

Pada *Test Run* nilai P-value dari quantumrandom lebih tinggi dan lebih stabil daripada nilai p-values dari SecureRandom. Pada quantumrandom nilai p-values tidak pernah 0, meskipun tidak pernah 1 juga, tapi ini sudah menunjukkan bahwa quantumrandom lebih stabil. Nilai rata rata dari p-values quantum random juga lebih tinggi dari SecureRandom, nilai rata rata ini

menunjukkan bahwa performa quantumrandom dalam menghasilkan angka acara lebih baik daripada Secure Random. Jadi dalam menghasilkan angka acak/random quantumrandom dan Random.org lebih baik daripada SecureRandom dan AppZaza. Dil

Penelitian ini dapat diperbaiki dengan melakukan analisis lebih dalam dengan menggunakan data yang lebih banyak dan range yang lebih luas. Juga bisa dilanjutkan dengan melakukan pengetesan dengan Test Suite lain yang ada pada NIST Test Suite.

References

[1] R. Munir, Slide Kuliah IF4020 Kriptografi, Pengantar Kriptografi, 2019.

[2] R, Andrew et al, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications , 2010.

[3] Stipčević, Mario & Kaya Koç, Çetin. (2014). True Random Number Generators. 10.1007/978-3-319-10683-0_12.

[4] The Distributed Systems Group, Computer Science Department, TCD. Random Number Generators:An Evaluation and Comparison of Random.org and some Commonly Used Generators, April 2005

[5] https://www.techopedia.com/definition/9091/random-number-generator-rng

[6] https://www.howtogeek.com/183051/htg-explains-how-computers-generate-random-numbers/

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Mei 2019

Ahmad Faishol Huda
13516094