Sinteza
Međunarodna naučna konferencija Univerziteta Singidunum

Impact of Internet on Business Activities in Serbia and Worldwide
Uticaj Interneta na poslovanje u Srbiji i svetu

DOI: 10.15308/SINTEZA-2014-606-609

ONE METHOD FOR GENERATING UNIFORM RANDOM NUMBERS VIA CIVIL AIR TRAFFIC

Milomir Tatović, Saša Adamović, Aleksandar Jevremović, Milan Milosavljević
Singudunum University, Serbia

Abstract:
In this paper we have used data from publicly available database of civil aviation flights worldwide shown in real time. For research purposes we have developed software for collecting and filtering data, that have originated from variables determined in a particular time and space that makes this dataset nonlinear. For the purpose of coding information source we have developed an algorithm for presenting data in bits based on which we encode received data and in that way we get binary sequences of sufficient length that present the basis for generating the cipher keys. We have exposed the obtained binary sequences to rigorous informational analysis, whose main purpose was to confirm the quality of the data according to NIST standards. After the analyses, we have confirmed the assumption that sequences obtained in this way can be used for cryptographic purposes, in the domain of generating a high entropy cipher keys.

Key words:
ADS-B,
Symmetric key generation,
TRNG.

INTRODUCTION

The necessity for data encryption does not represent a novelty. Generations of people have been trying to protect their communication in the safest way possible. With the development of computer and electronic communication, needs for data encryption are increasing. This has led to the development of a large number of algorithms for encryption which also affected the quality of algorithms and keys. Symmetric and asymmetric keys are used, depending on the design of encoding and decoding algorithms.. In addition to these coding algorithms there are algorithms that can provide perfect secrecy or absolute safety. One–Time pad is a perfect example which exactly represents a theoretically provable coding system. In order to provide perfect secrecy, this coding system requires the following conditions to be fully met: keys need to be generated completely at random (such resources may be found in nature), key length must be equal to the length of encrypted message that is being encrypted and the same key must never be used twice.

Our first task is generating random sequences. Random sequences can be generated by using TRNG and PRNG. TRNG represents a generator of truly random sequences that can be created in natural surroundings, while PRNG generates pseudo random sequences. These sequences are based on internal generator's state which has characteristics of truly random sequences. Both generators can be used for designing coding systems. Whether we use one or another, is necessary to generate truly random sequences with a unique role in the generating cryptology keys.

As a second task, we need identical random sequences in all communication points, bearing in mind the fact that perfect coding systems use the design of symmetrical coding systems. In case that we have to enable communication between physically distant points, we face a problem related to the distribution of cryptology keys. This case scenario is perfectly logical because cryptology key can be generated at a single point or on one side only. In practice, there is no secure communication channel, except "courier service".This service is responsible for delivering keys. However, the question arises as to what we should do in cases when this scenario is inapplicable.

In this paper we will try to provide a platform for synthesis of such a system that will solve both problems in the way acceptable for synthesis of perfect Coding Systems.

Contributions of this paper are numerous. Not only we provide cryptology keys of high quality and sufficient length, but we also create conditions for synthesis of systems for distribution of cryptology keys.

STATE OF THE ART

Besides a large number of quite reliable coding algorithms, a question arises as to "whether One-Time Pad is a

matter of past". If we find out a way to overcome problems mentioned in the previous chapter, the answer is obvious „absolutely no!".

Scientists tried to use crystal oscillators that proved to be extremely good generators of random sequences. Oscillators were placed on both sides of communication and thus produced identical random sequences. However, the problem arose from the need for occasional synchronization of oscillators, due to which this solution has not been widely accepted. A potential solution may come from protocols that are currently used for the exchange of cryptology keys. The most frequently used methods of key distribution are Diffie-Hellman protocol and key protection in public communication channels by using asymmetrical codes – RSA or DSA. Protocols resulting from quantum cryptography create basis for further development of protocols in classical cryptography. One of the most significant protocols among those in quantum cryptography is BB84. BB84 requires two communication channels. The first one is a one-way quantum channel, which represents an optical link between two participants that enables transfer of light (photons). The other one is a two-way public channel. Ueli Maurer realized the potentials of the second part of BB84 protocol and published a paper where he theoretically presented a new protocol, known as "Satellite Scenario". In this scenario, the first part of BB84 protocol requiring a one-way quantum channel was "replaced" with a satellite. Satellite Scenario is introduced in the way that requires a central source of randomly generated sequences (satellite). It is mandatory that each participant, "listening" to what the satellite is transmitting, receives a sequence with errors in different places compared to the original – transmitted sequence. This protocol enables exchange of a symmetrical key without any already known parameters. Apart from that it also does not theoretically limit the length of a key to be exchanged. Therefore, it is possible to use this protocol as a method of exchanging extremely long sequences – keys, which provides opportunity for perfect security.

METHODOLOGY

There are thousands of passenger and cargo airplanes above us at any moment. Talking about Europe only, at less frequent time the number of flights is approximately 1000. During the peak hours that number is several times larger. Nowadays, flights are tracked by GPS [3] navigation system. Knowing that civil aviation data are publicly available, we can precisely determine geographic latitude and longitude, speed and altitude of an airplane from any point on Earth. As a consequence of airplane high speed and GPS accuracy, coordinates are changed very fast in a unit of time. Therefore the possibility that an aircraft is placed at the same geographical location at the same moment is only theoretical. However, if we possess information about several flights at the same moment, we eliminate a theoretical possibility that the value of geographical location of all those flights is predictable in a unit of time. On the basis of this assumption we have analyzed data related to flights from different locations. The advantage of this system is that it enables access to the same data from two physically separated computers. With this adventage the system can provide distribution of cryptology keys among several users.

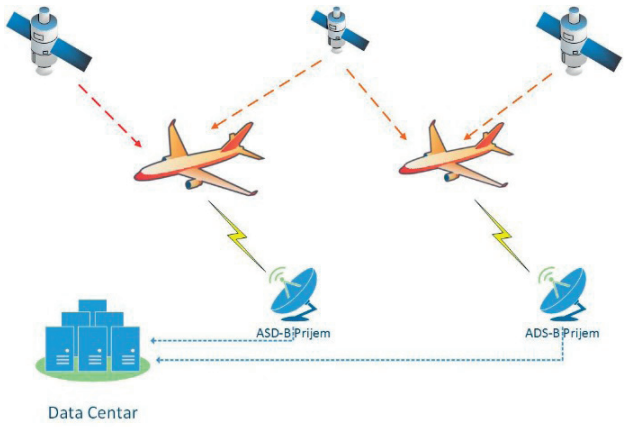

Fig. 1. ADS-B

As it is shown in Figure 1, GPS satellites communicate with airplane devices, thus providing data necessary for flying the plane. Further on, airplanes take over the satellite role and transmit a new radio signal including data relevant for that flight. These data are collected in air traffic control centers. According to civil aviation laws they can be collected by all interested parties using ADS-B [4] technology. This method does not allow data collection from a single location for the whole Europe. That's why data are collected from numerous locations and stored in Internet databases in order to be used for real time monitoring of airplanes. Figure 2 shows one of these services that we are going to use By means of an application specially created in Java programming language, we have enabled both sides participating in safe communication to receive air traffic dat. After receiving the air traffic data, they are filtered and coded with the aim of gaining long binary sequences.

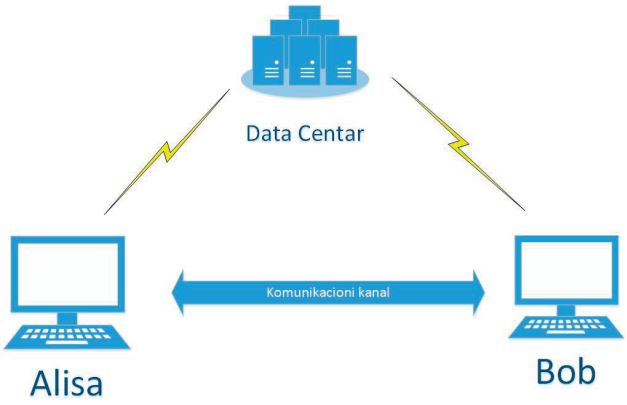

Fig. 2. User - Database communication

Data collecting and processing – signal processing

After the data have been downloaded from the central database we proceed to the second phase – filtering. Filtering phase implies ignoring data about aircrafts that are still in the stage of landing or taking off. The second phase also includes data selection. Database stores geographic latitude and longitude, altitude and flight speed, informa-

tion on flight number, arrival and departure airport and airlines. We only select those particular data which value is changed in a unit of time. By carried out analysis we have determined that the only relevant variables are those related to geographic latitude and longitude, while speed and altitude have constant values. We come to the last process in the filtering phase. The data on geographic latitude and longitude are presented by real two digit numbers with four decimal places. As a result of high accuracy in decimal input value, numbers preceding the decimal point do not have instant modification. Because of that, those numbers are also eliminated and only decimal values are left. In the analysis result shown in Figure 3 (on the left), it is clear that due to decimal rounding the values ranging from 0000 to 0500 and from 9500 to 9999 are more frequent than those from other intervals. This is the reason why we have excluded these two ranges (Figure 3 on the right). This anomaly can be interpreted in a different way as an error visible in our system resulting from rounding real numbers with four decimal places.

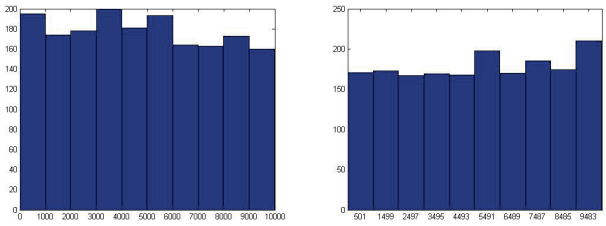

Fig. 3. Optimization of values from set intervals

Coding of information source (ADS-B service)

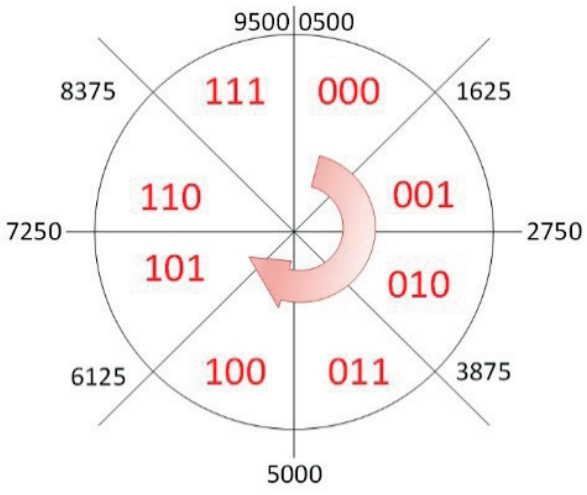

Fig. 4. Phasor

Following the data filtering phase, it is necessary to transcode data into the form of binary array. This method of signal processing is known as binarisation. This phase has been named information source coding phase. The value of geographic latitude and longitude coordinates will be expressed by three bits so that each flight will be shown as six bit array. Using phasor shown in Figure 4 we perform coding in which the decimal value of geographic latitude or longitude gets a relevant bit representation. For example, in case that we track a flight above Belgrade, at the moment of flying over the Avala Tower, the airplane will have coordinates 44.6961 and 20.5144. As we have already mentioned, after filtering and data processing only decimal values 6961 and 5144 would be left. Phasorbit representation of this flight would be: 101100.

By its design the phasor shown in Figure 4 corresponds with one-dimensional Gabor filter. Coding phase has been conducted using data normalized with 1D Log-Gabor filter. By each filter we get 6 bit data array from a single phasor part. Each phasor output is set in the way that only one bit is changed while going from one into another. This type of quantification will provide more consistent information as it reduces correlation among inputs.

After the coding phase of each flight and expressing each of them in the form of six bit array we get a sequence of at least six thousand bits per iteration of communicating with the central database. After 15ms which is the time needed for database updating, the procedure is repeated and we get a new sequence of six thousand bits and so on. Due to the Internet connection speed we are not able to follow the movement of a particular aircraft every 15ms. This fact helps us get drastic change of coordinate values. In this way we can get a large number of binary arrays in time interval of a few seconds. However, a number of bits per second produced by a TRNG is several times larger, but the advantage of our system is the fact that TRNG can produce bits only on one side of secure communication. Using this method we generate identical binary arrays at different locations with the aim of achieving secure communication.

EXPERIMENTAL DATA ANALYSIS

In order to prove the initial assumption that bit arrays generated in this way have a high level of randomness, we have used statistical tests required by US National Institute of Standards and Technology (NIST). These tests are published and defined in paper 800-22 from April 2010, named "A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications". The tests have been developed for testing binary sequence randomness by focusing on different types of steadiness that might exist in a sequence. A package of tests consists of 15 different statistical tests, some of which are divided into subtests.

For our experiment we selected most commonly used tests with highly reliable results. In order to get a comparative test with already proven TRN generator we used data published on website random.org. Those data are considered to be truly random.

Testing frequency in a sequence (Frequency test)

The aim of this test is to analyze relationships between 1 and 0 in a bit array. It is necessary to have approximately the same number of 1 and 0 in a sequence.

Table. 1. Frequency test results

	random.org	Our sequence
P-value	0.808976294576234	0.8146073609727413

The number of bits used in this test as well as in subsequent tests is 9856. After conducting the test we can conclude that both sequences met the requirement set by NIST

$$P \geq 0{,}01 \quad (1)$$

and they can be considered random. If we compare results we come to the conclusion that 1 to 0 ratio in both sequences was approximately the same.

Serial test

This test is used for determining frequency of all potential overlaps of n-bit array in the whole sequence.

Table 2. Serial test results

	random.org	Our sequence
Trigrams		
000	1215	1208
001	1250	1209
010	1280	1157
011	1195	1225
100	1251	1209
101	1225	1173
110	1195	1125
111	1245	1215
P-val 1	0.3174364812591314	0.6440496128821558
P-val2	0.1088774362443912	0.4608606318159197

The results of this test meet the NIST requirement defined in formula (1). Therefore, on the basis of this test we can also state that sequences are random, noting that the test with our sequence showed a more stable 1 to 0 overlap ratio.

Runs test

The characteristic examined in this test is the total number of consecutive repetition of 1 or 0 in a sequence.

Table 3. Runs test results

	random.org	Our sequence
P- Value	0.6426910024891495	0.353825079052692

The results of this test meet the requirement defined in the formula (1) and we consider sequences to be random. By comparing these two sequences after the test we can conclude that the one taken from website random.org has more changes from 0 to 1 or vice versa.

Entropy test

In this test we examine occurrence frequency of all potential overlapping n-bit samples in a sequence.

Table. 4. Entropy test results

	Random.org	Our sequence
Monobit	0.9999957227359038	0.9999958775044104
Bigram	0.9999878164470851	0.9999625655234716
Trigram	0.9998791930081991	0.9999120107423237
4x4 Matrices	0.9998558732137256	0.9997868840346474

This test values also meet NIST requirements formula (1). We can state that our sequence is random and compared to the sequence taken from website random.org it shows better results. Accordingly we can state that it has a high level of randomness as well as that it can be used for generating high quality cryptology keys.

CONCLUSION

On the basis of experimental analysis and the results obtained, we conclude that our initial assumption is correct. As a result, this method can provide high quality source of information consisting of truly random binary arrays. However, for generating high quality cryptology keys we need a protocol to provide generating identical cryptology keys on two sides based on the information source available to all participants. Theoretical protocol "Satellite Scenario" already mentioned in Chapter two may be one of the possible solutions. Considering the fact that protocol "Satellite Scenario" is known only in theory and belongs to the category of perfect protocols, the satellite role in our realization would be taken by ADS-B service. This service acts as a collector of data available to all participants in communication at any point on Earth.

A future paper would be based on designing a complete encryption system. This system would have implemented module for generating a cryptology key, inspired by the idea presented in this paper. This way we could enable a regular use of the most secure encryption system - One-Time pad, that provides absolute computer safety.

REFERENCES

[1] C. Shannon, Communication Theory of Secrecy Systems, Bell Systems Internal publications.

[2] U. Maurer, Secret Key Agreement by Public Discussion, IEEE Trans. INFORM. Theor. 39 (1993) 733-742.

[3] "The Global Positioning System: A Shared National Asset", National Academies Press, 1995. (*references*).

[4] Christine Vigier, "Automatic Dependent Surveillance Broadcast", AirBus. (*references*).

[5] Rukhin, Andrew, Soto, Juan / Nechvatal, James. A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications. [pdf] 2010.