Quantum1Net

Introducing a Non-Algorithmic, Computationally Irreducible, Scalable Data Encryption Framework

Introduction

In this paper, we present a brief overview of a new kind of Quantum Encryption Key Generator (QEKG) apparatus, designed to generate a continuous stream *of fundamentally random sets of numbers with certain intrinsic mathematical properties because of quantum nature of the generator*. QEKG-produced random numbers can be used as a set of mutually authenticated cryptographic keys of arbitrary length, significantly reducing computational overhead associated with encryption/decryption of individual data objects and distribution of cryptographic keys. Because encryption keys composed of random numbers that are generated utilizing quantum entanglement, there is no computing algorithm associated with such keys. Utilizing such encryption keys for data encryption makes the decryption process without the original keys by a brute force or other means computationally irreducible, meaning that no shortcuts exist that would allow it to be performed more rapidly. While quantum computing attacks against all mainstream public-key algorithms including RSA, Diffie-Hellman and elliptic curve cryptography utilizing Shor's or algorithm or its derivatives are widely conjectured to be effective, they will not be effective against the encryption keys generated using QEKG.

Overview

Generating fundamentally random numbers is absolutely vital in such applications as cryptography, computer security, numerical simulation, scientific research, and the gaming industry. Commonly used software-based pseudo-random generators are not quite random and have other well-known deficiencies. Commercially available Quantum Random Number Generators (QRNG), which recently appeared on the market, by their nature produce a fundamentally random number. A good overview of quantum number generations can be found in [1]. This paper also covers critical application differences between software-based pseudo-random generators and sources of fundamentally random numbers like QEKG.

Whatever the source is, these values combined with very specific properties of selected mathematical objects are being used in modern cryptography, as well as in cyber-security and data protection systems as the basis for computing complimentary encryption keys and similar purposes. This approach is not problem-free; in most practical instances, it requires significant computational overhead and it has known vulnerabilities.

A critical difference between the proposed Quantum Encryption Key Generator (QEKG) and is that QEKG is generating a stream *of fundamentally random sets of numbers with certain intrinsic mathematical properties as a consequence of quantum nature of the generator*. QEKG-produced random numbers can be used as a set of mutually authenticated cryptographic keys of arbitrary length, significantly reducing computational overhead associated with encryption/decryption of individual data objects and distribution of cryptographic keys.

In addition, encryption key sets generated by QEKG can be utilized as an extra layer of security and authentication, making existing digital certificate systems, Public Key Infrastructures (PKI), and data transmission protocols more resistant and in some instances even virtually immune to tampering, eavesdropping and interception.

Quantum1Net have developed a QEKG workbench system, which is based on a one-qbit quantum optical device, which utilizes quantum entanglement phenomena. It uses a pair of Type-I SPDC-generated entangled photons measured by four photon detectors and time-to-digital (TDC) converter to generate random sets or numbers associated with unique timestamps.

The CW 404nm laser pump and a Type-I BiBO crystal are used as an entangled photons source. Idle and signal photons are passed through ¼ wave plate (QWP) and polarizing beam splitter (PBS) each to prepare the quantum system in the desired state. The resulting wave functions are measured by single photon detectors, producing TTL pulses which are assigned unique timestamps by the 4-channel time to digital converter (TDC). The output of the TDC is read by a classical computer, converted to sets of encryption keys using a variable coincidence count window which are stored in the temporary FIFO queue. Resulting encryption keys have intrinsic mathematical properties defined as multiplicative cyclic group of order 2N (denoted as Z_{2N}) and streamed using industry standard RTP and RTSP protocols on four separate channels. To securely transmit data between 2 communicating clients, the transmitter accesses any two of the streams and forms an encryption key to encrypt the data before sending to the receiver. The receiver accesses 2 renaming streams of random numbers to form a decryption key. The timestamps associated with each stream and information about the width of the coincidence window is used to synchronize receiver and transmitter encryption keys to make the data decryption possible.

Lab Prototype and Statistical Tests of Randomness

Based on our previous experiments of the qualitative analysis of BBO and BiBO Crystals [4], we decided to utilize 5x5x3mm Type-I BBO crystal in our first prototype to produce entangled photons that can be used to generate random numbers by measuring coincidences and amplitudes of single photons in a specified period of time.

The fundamental process behind using BBO crystals for entanglement is spontaneous parametric down-conversion (SPDC). SPDC occurs when polarized photons from a laser are passed through a crystal with certain properties. In the case of BBO, this crystal is ß-barium borate (ß-BaB2O4). BBO is a strongly birefringent and optically nonlinear crystal in the x and y axes.

SPDC occurs when the electric field of the incoming laser beam interacts with the dipole moment of the molecules in the crystal, and produces the following result:

when a photon from a laser beam (the “pump” photon) enters the crystal, it is split into two entangled photons (the “signal” and “idler” photons), each with twice the wavelength (half the energy) of the original photon. Additionally, these photons are phase-matched in frequency and are in a state of polarization superposition. Different types of crystals put these photons into different states.

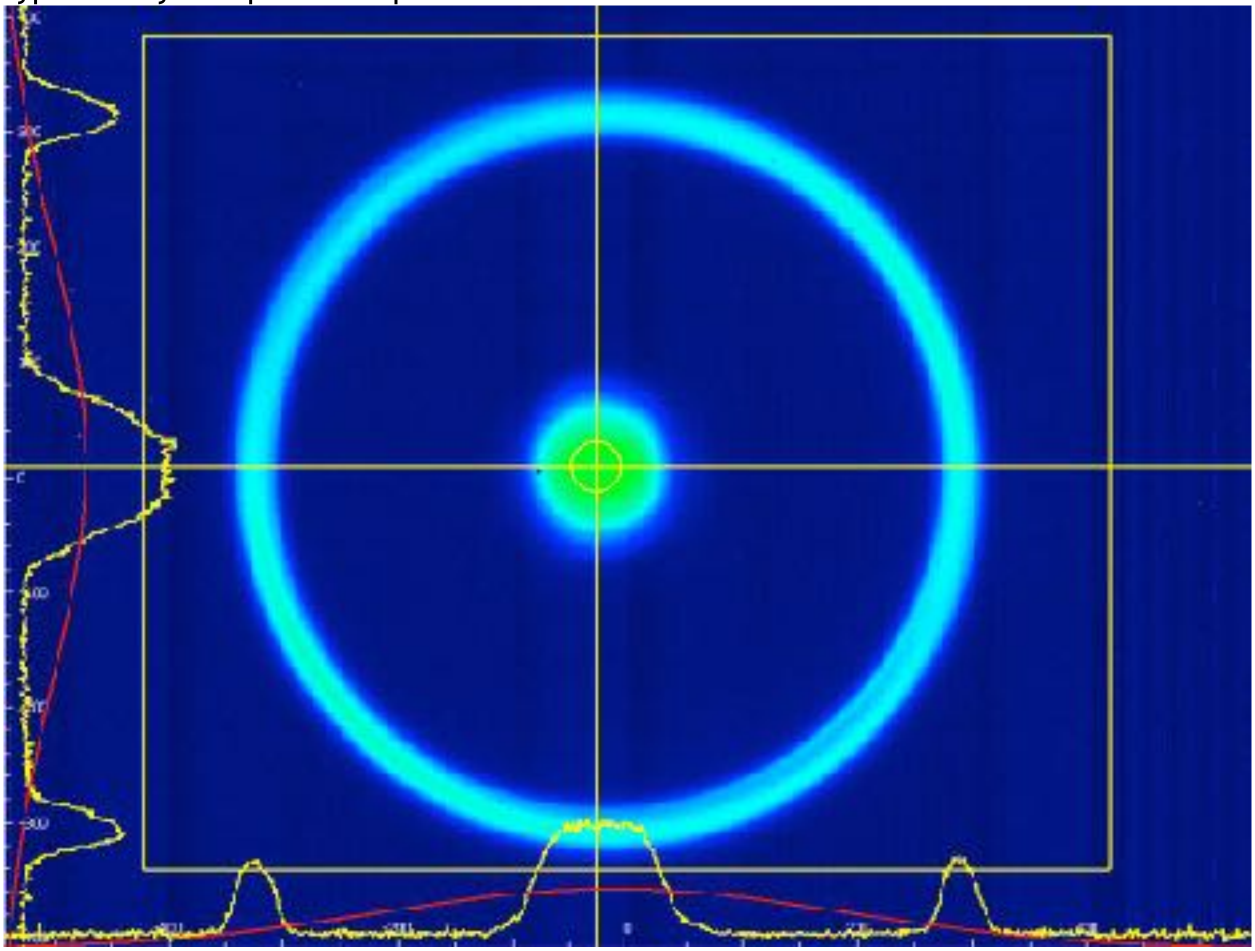


Figure 1 The cone of polarization-entangled photons produced by Type-I BBO crystal can be seen on beam analyzer camera. (Note: center bright spot corresponds to the residual pump beam).

Type I crystals produce a single cone of photons (figure 1) with entangled photons on opposite sides of the cone. In Type I entanglement, the signal and idler photons are both polarized on what is called the ordinary axis (as opposed to the extraordinary axis). In other words, if the signal photon is measured to be polarized vertically, we immediately know that the idler photon is also vertically polarized. SPDC photons are also spatially entangled, meaning that if we detect the signal photon at a specific space coordinates; we immediately know where the idle photon is at the same time.

Theory of QEKG

Type-I SPDC produces a pair of entangled photons with the same polarization we denote as $|V_s\rangle$ and $|V_i\rangle$ for signal and idle photons respectively. The wave function then can be written as

$$|\gamma\rangle = \frac{1}{\sqrt{2}}\left(|V_i\rangle + |V_s\rangle\right)$$

We are interested in the expectation values (amplitudes) for each detector and coincidence counts for $\langle D1\rangle\langle D3\rangle$, $\langle D1\rangle\langle D4\rangle, \langle D2\rangle . \langle D3\rangle$, $\langle D2\rangle\langle D4\rangle$ detector pairs

$$\begin{aligned}
\langle D1\rangle &= \left\langle V_i \middle| (HpQwp)^{\dagger} HpQwp \middle| V_i \right\rangle \\
\langle D2\rangle &= \left\langle V_i \middle| (VpQwp)^{\dagger} VpQwp \middle| V_i \right\rangle \\
\langle D3\rangle &= \left\langle V_s \middle| (VpLCwp)^{\dagger} VpLCwp \middle| V_s \right\rangle \\
\langle D4\rangle &= \left\langle V_s \middle| (HpLCwp)^{\dagger} HpLCwp \middle| V_s \right\rangle
\end{aligned}$$

Here, the polarizing beam splitter is defined as (for horizontal and vertical polarizations respectively):

$$\begin{aligned}
Hp(\tau, r) &= \frac{1}{\sqrt{1-\tau}}\begin{pmatrix} r & 0 \\ 0 & 1-\tau \end{pmatrix}, \quad 0 \le r \le \tau \ll 1 \\
Vp(\tau, r) &= \frac{1}{\sqrt{1-\tau}}\begin{pmatrix} 1-\tau & 0 \\ 0 & r \end{pmatrix}, \quad 0 \le r \le \tau \ll 1,
\end{aligned}$$

where $r = \tau = 0$ 0 for an ideal PBS, however for a real PBS it is proportional to transmission and reflection efficiency and may have different value for Hp and Vp;

the Qwp is a Quarter Wave Plate:

$$Qwp(\vartheta) = \begin{pmatrix} \cos^2\vartheta + i\sin^2\vartheta & (1-i)\cos\vartheta\sin\vartheta \\ (1-i)\cos\vartheta\sin\vartheta & i\cos^2\vartheta + \sin^2\vartheta \end{pmatrix}$$

LCwp is a Liquid Crystal wave plate:

$$LCwp\left(\vartheta, \varphi\right) = \begin{pmatrix} \cos^2\vartheta + e^{i\varphi}\sin^2\vartheta & (1-e^{i\varphi})\cos\vartheta\sin\vartheta \\ (1-e^{i\varphi})\cos\vartheta\sin\vartheta & e^{i\varphi}\cos^2\vartheta + \sin^2\vartheta \end{pmatrix}$$

φ is a retardance, which depends on the applied voltage. When $\varphi = \pi/2$ the LCwp equation becomes Qwp.

For an ideal PBS ($p \;\; = 0$) the resulting equations for coincidence count amplitudes of four possible detector combinations, with the quarter wave plate's fast axis positioned at 45 degrees, are as follows:

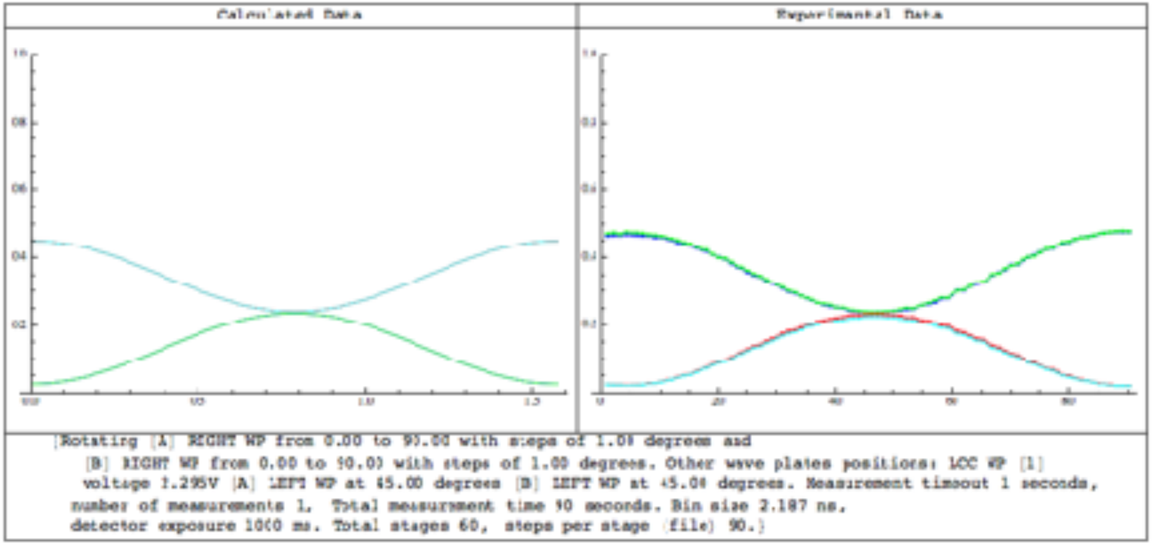
$$\langle D1 \big| D3 \rangle = \langle D1 \big| D4 \rangle = 2\cos^2\theta \sin^2\theta \sin^2\frac{\varphi}{2}$$

$$\langle D2 \big| D3 \rangle = \langle D2 \big| D4 \rangle = \frac{1}{8}(3 + \cos 4\theta + 2\cos\varphi \sin^2 2\theta)$$

We are interested in the point where all expectation values are equal, which corresponds to equal probability of registering a coincidence count between any one of four detector pairs (figures 2). Positioning both wave plates at 45 degrees leaves one degree of freedom φ. By slightly varying LCwp voltage, the retardance φ we can be slightly adjusted to ensure that the physical apparatus does deviate from the ideal alignment.

$$\langle D1 \big| D3 \rangle = \langle D1 \big| D4 \rangle = \frac{1}{4}(1 - \cos\varphi)$$

$$\langle D2 \big| D3 \rangle = \langle D2 \big| D4 \rangle = \frac{1}{4}(1 + \cos\varphi)$$



[Rotating [A] RIGHT WP from 0.00 to 90.00 with steps of 1.00 degrees and [B] RIGHT WP from 0.00 to 90.00 with steps of 1.00 degrees. Other wave plates positions: LCC WP [1] voltage 1.295V [A] LEFT WP at 45.00 degrees [B] LEFT WP at 45.00 degrees. Measurement timeout 1 seconds, number of measurements 1. Total measurement time 90 seconds. Bin size 2.187 ns, detector exposure 1000 ms. Total stages 60, steps per stage (file) 90.]

Figure 2 This graph shows comparison of experimental coincidence counts between detectors ⟨D1 ∣ D3⟩, ⟨D1 ∣ D4⟩, ⟨D2 ∣ D3⟩, ⟨D1 ∣ D4⟩ (left) and theoretical expectation values (right). We are interested in the point where all 4 curves intersect. LCWP angular position (from 0 to 90deg.) is plotted on the horizontal axis. Normalized coincidence counts are plotted on the vertical axis. QWP is fixed at 45 degrees. LCWP voltage is set to 2.295V.

We can now use coincidence counts for detector pairs $\langle D1 \big| D3 \rangle$, $\langle D1 \big| D4 \rangle$, $\langle D2 \big| D3 \rangle$, $\langle D1 \big| D4 \rangle$ to generate sets of encryption keys of arbitrary length by encoding this data as 4 orthogonal vectors, such as {1,0,0,0}, {0,1,0,0}, {0,0,1,0}, {0,0,0,1}. These vector sets form cyclic group of order 4 called Z_4 with group element $\langle i \rangle$: (i^n, n={1,2,3,4}). Properties of this group are suitable for implementing new encryption algorithms, discussed below.

Prototype Implementation

For convenience, we have built our prototype using large optical components on an optical workbench (figure 3). It is important to note, however, that it will be significantly reduced in size to fit in standard computer case, even on a standard computer board. The prototype uses quantum entanglement phenomenon to generate random numbers by recording single photon arrival times using 4 Single Photon Counting Modules (SPCM) and a Time-to-Digital Converter (TDC).

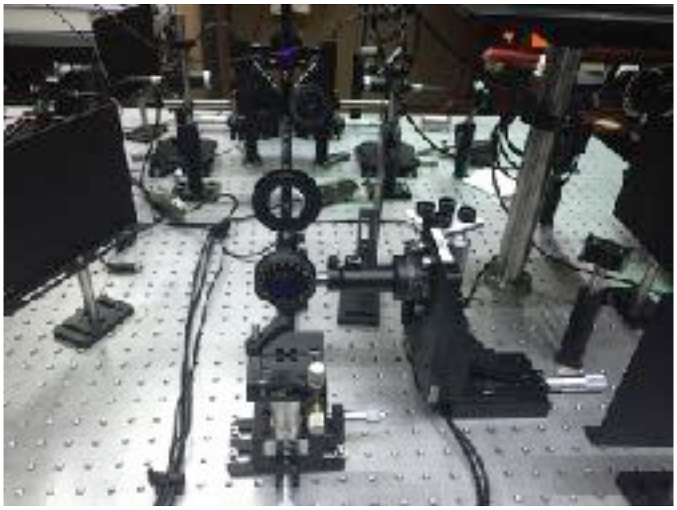


Figure 3 Prototype implementation. Using proper filters it is possible to run this experiment with bright white LEDs (as pictured) without affecting measurements.

In our workbench version, we use ~70mW 404nm (blue ray) laser beam passing through a ½ -wave plate and 5x5x3mm Type-I BBO crystal produces a pair of polarization-entangled photons with probability P ~10^{-11}. The *idle* photon passes though 800nm long pass filter, a ¼ wave plate positioned at 45 degrees, and a polarizing beam splitter (PBS). Photon arrival times are measured at each PBS exit by a fiber coupled single photon detectors (SPD1, and SPD2). The *signal* photon passes through another 800nm long-pass filter, a liquid crystal optical retarder (a wave plate with adjustable retardation angle, depending on the applied voltage) rotated to 45 degrees, and another PBS. Two more photon detectors (SPD2 and SPD3) are placed at each output of the second PBS (figure 4).

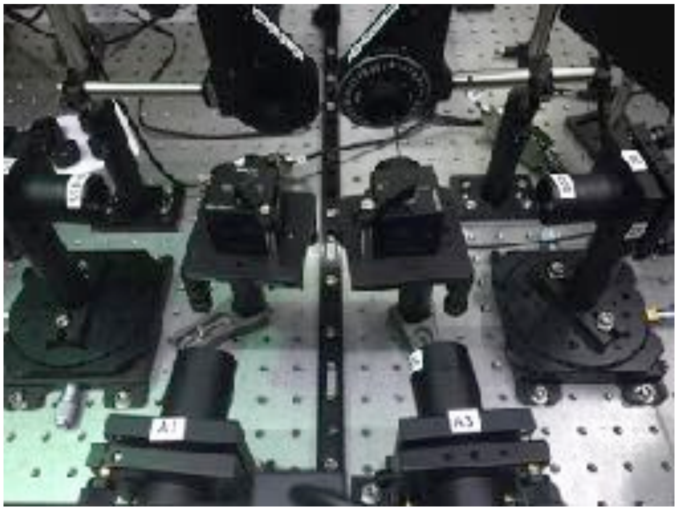
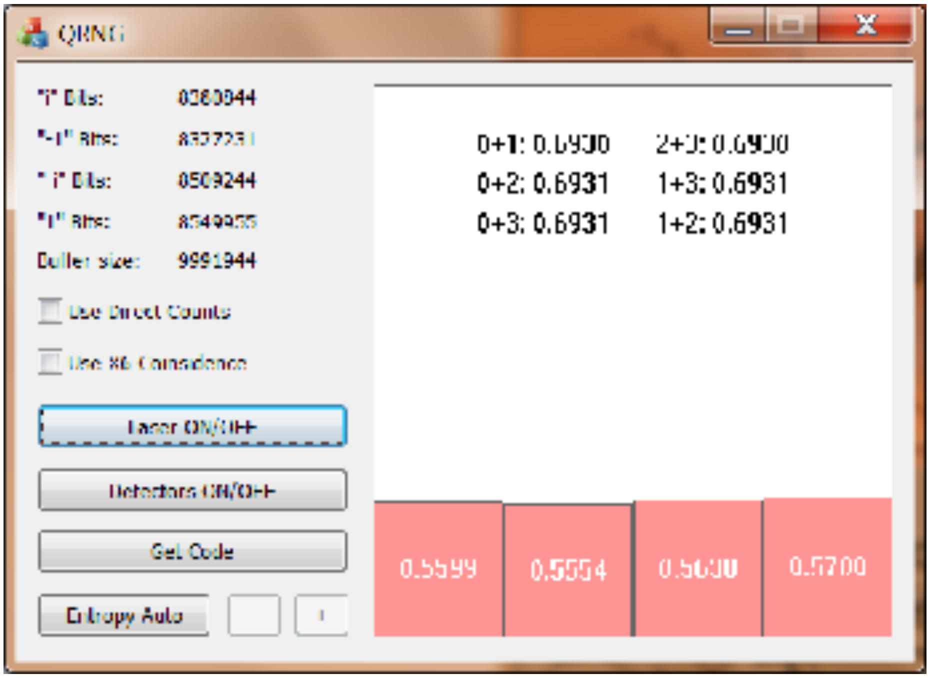


Figure 4 Two PBSs (center), ¼ wave plate (left), Liquid crystal wave plate (right) and fiber couplers (fiber-coupled

The output of each PBS is filtered using an additional band-pass 800±10nm filter, then directed to the SPCM-AQ4C Single Photon counting Module 4 Channel Array by Pacer, using 1m multimode fibers. SPCM produces TTL pulses that correspond to a single photon detection. These pulses are analyzed by the ID800-TDC time-to-digital converter by Quantique (in this configuration only 4 first channels are used).

TDC records time of arrival of each photon (TTL pulse) on each of its channels with resolution of 81ps. A simple computer program (figure 5) reads the TDC output via USB and records coincidence and amplitude counts. We use 2.4ns coincidence window.



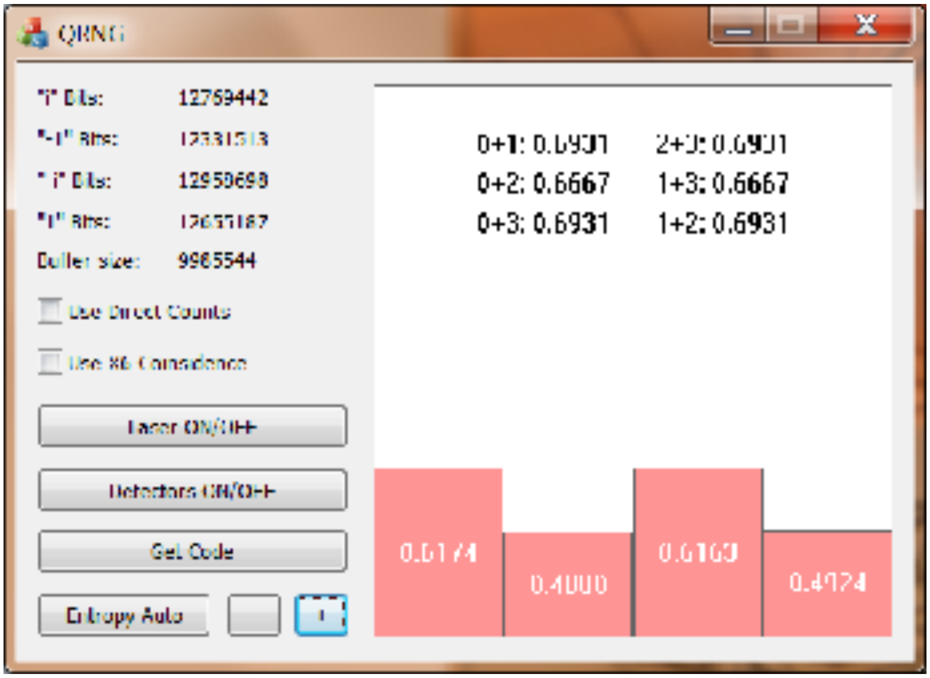
```
"I" Bits:       8380844
"-I" Bits:      8327231
" i" Bits:      8509244
"i" Bits:       8549955
Buffer size:    9991944

0+1: 0.6930    2+3: 0.6930
0+2: 0.6931    1+3: 0.6931
0+3: 0.6931    1+2: 0.6931

0.5599   0.5554   0.5630   0.5700
```

Figure 5 QRNG program in automatic mode computes histogram and entropy corresponding to coincidence counts for each “half-key” and 3 possible full keys (with their inverse).

This prototype generates fundamentally random numbers by comparing time stamps (TS) between detector pairs $\langle D1 \big| D3 \rangle$, $\langle D1 \big| D4 \rangle$, $\langle D2 \big| D3 \rangle$, $\langle D1 \big| D4 \rangle$. When TS are within the confidence window for a specific detectors pair, new column vector is recorded with 1 in the position of the confidence and 0 in remaining 3 positions. TS for all for detectors that correspond the coincidence event are also recorded. Time stamps are later used to align encryption keys. The encryption protocol uses time stamps to authenticate communicating clients by sending TS from detectors $\langle D1 \big| D2 \rangle$, along with 2 “half-keys” to the receiver, and TS from $\langle D3 \big| D4 \rangle$ and remaining 2 “half-keys” to the transmitter. To make the communication even more secure, individual “half-keys” and TS are sent using different networks, network subnets, and/or network protocols. During the secure data transmission, time stamps are compared between the transmitter and the receiver to verify that the same time stamps have been used to generate encryption keys. Important to note that time stamps are also fundamentally random numbers, while linked to encryption keys by width of the coincidence window, which provides additional layer of authentication of communicating devices. For example, if, during the secure data transfer, the encryption “half-key” #1 contains bit “1”, which corresponds to the coincidence of the $\langle D1 \big| D3 \rangle$ detector pair, but TS from this detector pair are not within coincidence window, the communication is not authenticated and should be terminated.

The rate generated random numbers is proportional to the laser power and characteristics of the BBO crystal. For instance, using 70mW 404nm laser and 3mm thick Type-I BBO crystal, random number rate is ~250Kbit/second using coincidence counts and ~2Mbit/second using amplitude counts. This rate can be easily increased by an order of magnitude. For instance, see [6] about optimizing type-I polarization-entangled photons and other useful references on the subject.

The liquid crystal wave plate is controlled by the PID algorithm to ensure even (while still fundamentally random) distribution of counts for all four detectors (each full key, which is a sum of any 2 “half-keys” has entropy of Log(2)~0.6931). Alternatively, using a varying voltage on the liquid crystal wave plate can be used to induce additional entropy in the output results, to give the QEKG a unique “character” that can be used as additional security measure, because it cannot be commuted or simulated using a classical computer. The QRNG program provides UI to manually set liquid crystal wave plate voltage (see figure 6).



"i" Bits: 12769442
"-1" Bits: 12331513
"-i" Bits: 12950698
"1" Bits: 12633187
Buffer size: 9985544

□ Use Direct Counts
□ Use X6 Coincidence

0+1: 0.6931 2+3: 0.6931
0+2: 0.6667 1+3: 0.6667
0+3: 0.6931 1+2: 0.6931

0.6174 0.4000 0.6163 0.4724

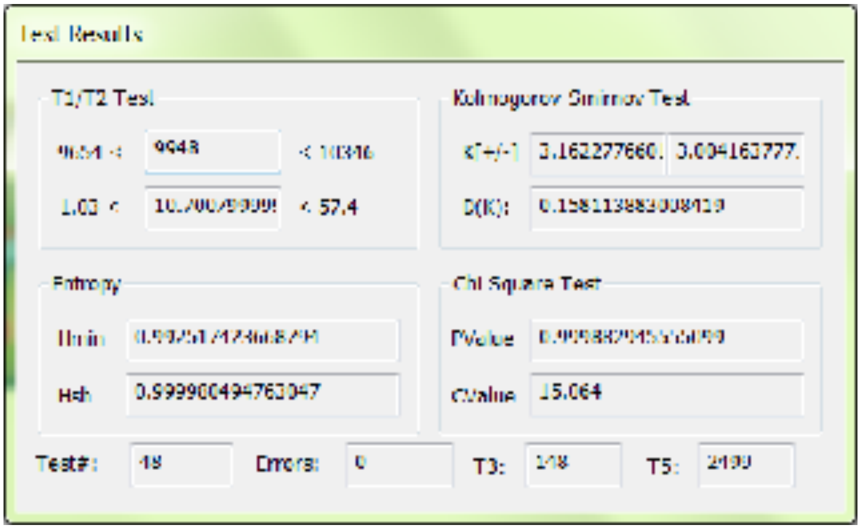
Figure 6 QRNG program in manual mode allows to programmatically changing distribution of bits between subsets while preserving fundamental randomness.

Statistical Tests of Randomness

The AIS 20 / AIS 31 document published by Bundesamt für Sicherheit in der Informationstechnik (BSI) titled Functionality classes for random number generators [7] provides a good classification of RNG and introduces a series of statistical tests to test true randomness of physical (ideal) RNG. The paper suggests 8 statistical tests to analyze random data generated by a physical RGN. The test criteria is set to assure that no deviation of the RNG from the ideal RNG can be found or used in practical attacks.

We remind the reader, that "…if the statistical test fails, the Null hypothesis is rejected. If the test value is not very unlikely, this does not confirm the Null hypothesis. Statistical tests cannot confirm the Null hypothesis. But the absence of evidence is not evidence of absence. The statistician decides whether he continues or stops testing on the basis of the number of conducted tests". In our case, the null hypothesis is that the data generated by the QEKG is indistinguishable from the data generated by an ideal RNG.

QRNG program samples random 20K-bit-long data chunks once per second and performs tests T1 - T8. Test results are dynamically displayed on the screen (see Figure 6) and also stored in the log file for future analysis. So far, we have accumulated more than 60 hours of continuous statistical data. All tests have passed within the specified parameters to qualify for an ideal RNG with a deviation error for all any specified test less than 2•10-8.



Test Results

T1/T2 Test

9654 < 9948 < 10346

1.03 < 10.7007999! < 57.4

Kolmogorov Smirnov Test

K[+/-] 3.162277660. 0.004163777.

D(K): 0.158113883008419

Entropy

Hmin 0.99251742868749

Hsh 0.999900494763047

Chi Square Test

PValue 0.9958829455560599

CValue 15.064

Test#: 48 Errors: 0 T3: 148 T5: 2499

Figure 7 Dynamic test results.

Supplemental materials for this article contain data for all statistical tests and a Wolfram Mathematica notebook to visualize and analyze results. We performed the above tests using data from QEKG as well as data from commonly used Pseudo Random Number Generators (PRNG), such as rand() function in C++. Charts on figures 8 and 9 show results of Chi Square (P-value) and Entropy distribution calculations performed using 50,000 unique samples, each 20,000 bits in length. Charts on the left are based on results produced by the classical PRNG. Charts on the right show statistical analysis of experimental results from QEKG.

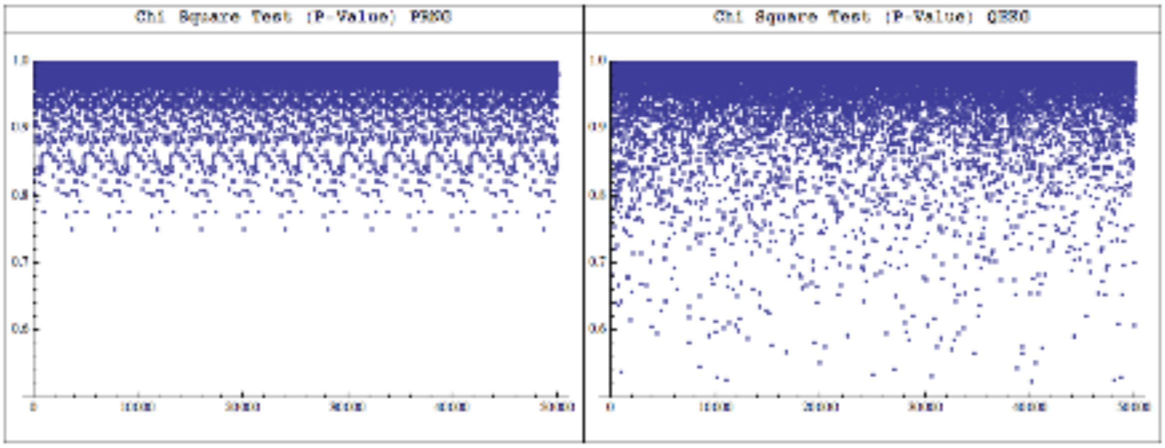


Figure 8 Chi Square (p-value) tests of 50,000 samples 20,000 bits in length generated by PRNG (left) and QEKG lab prototype (right).

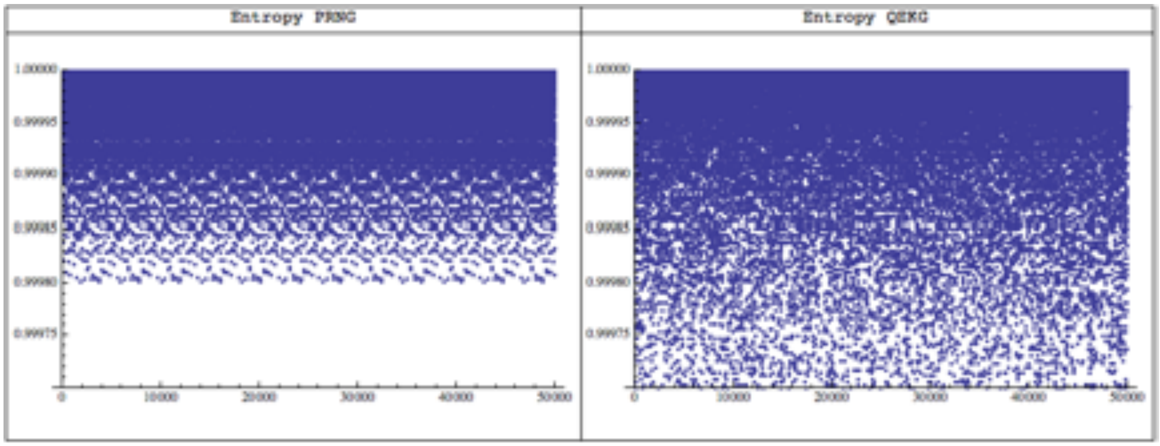


Figure 9 Entropy distribution of 50,000 samples 20,000 bits in length generated by PRNG (left) and QEKG lab prototype (right).

Just by visually comparing charts above, it appears that, unlike PRNG, QEKG has no obvious repetitive pattern. This makes it impossible to predict future QEKG data, regardless of how much previous data is analyzed. In contrast, PRNG exhibits an obvious pattern, making it possible to predict future data with a high accuracy if a large enough sample can be analyzed. This difference between PRNG and QEKG makes the latter most suitable for generating encryption keys.