sensors

MDPI

Article

Design and Analysis of a True Random Number Generator Based on GSR Signals for Body Sensor Networks

Carmen Camara [1,*,†,‡], Honorio Martín [2,‡], Pedro Peris-Lopez [1,‡] and Muawya Aldalaien [3,‡]

1 Department of Computer Science, University Carlos III of Madrid, 28911 Leganes, Spain; pperis@inf.uc3m.es
2 Department of Electronic Technology, University Carlos III of Madrid, 28911 Leganes, Spain; hmartin@ing.uc3m.es
3 Higher Colleges of Technology, Abu Dhabi Women's College, Abu Dhabi 41012, United Arab Emirates; maldalaien@hct.ac.ae
* Correspondence: macamara@pa.uc3m.es; Tel.: +34-916246260
† Current address: Department of Computer Science, University Carlos III of Madrid, Avda. de la Universidad, 30, 28911 Leganés, Madrid, Spain.
‡ These authors contributed equally to this work.

Received: 24 March 2019; Accepted: 23 April 2019; Published: 30 April 2019

Abstract: Today, medical equipment or general-purpose devices such as smart-watches or smart-textiles can acquire a person's vital signs. Regardless of the type of device and its purpose, they are all equipped with one or more sensors and often have wireless connectivity. Due to the transmission of sensitive data through the insecure radio channel and the need to ensure exclusive access to authorised entities, security mechanisms and cryptographic primitives must be incorporated onboard these devices. Random number generators are one such necessary cryptographic primitive. Motivated by this, we propose a True Random Number Generator (TRNG) that makes use of the GSR signal measured by a sensor on the body. After an exhaustive analysis of both the entropy source and the randomness of the output, we can conclude that the output generated by the proposed TRNG behaves as that produced by a random variable. Besides, and in comparison with the previous proposals, the performance offered is much higher than that of the earlier works.

Keywords: Galvanic Skin Response (GSR); entropy; randomness; Random Number Generators (RNG); Hilbert transform

1. Introduction

The proliferation of wearable sensors has meant that medical environments are not the only ones in which the acquisition of vital signs can occur [1]. For instance, there are a large number of smart-watches (or sports watches) that monitor several of our physiological signs throughout our daily lives, and even smart-textiles that have one or more integrated sensors have appeared on the market [2]. Concerning the measured signal, there is a wide variety ranging from signals related to the brain (e.g., Electroencephalogram (ECG)) through signals linked to the heart (e.g., Electrocardiogram (ECG) or Photoplethysmogram (PPG)) to signals related to emotions (e.g., Galvanic skin response (GSR)). Sensors do not usually work in isolation but form a network. When we refer to sensors that are in (e.g., a pacemaker or a neurostimulator) or around (e.g., an insulin pump or a sport-watch) the body, this type of network is named Wireless Body Area Network (WBAN) [3,4]. Body Sensor Network (BSN) or Medical Body Area Network (MBAN) are other names given to these networks [5,6]. Apart from the sensors, there is a central element called the gateway—a smart-phone usually implements the latter. Currently, the sensors do not communicate directly with each other (shortly this may happen),

but all connections pass through the gateway. It is also the gateway that provides connectivity to the Internet [7].

1.1. Related Work

In the context of cybersecurity, vital signs have proved to be very useful in recent years. Biometrics solutions based on ECG [8,9] or EEG signals [10,11] have been proposed for authentication purposes. Some authors have even studied its feasibility (ECG [12,13] or EEG [14,15]) in the context of continuous authentication—the verifier validates the credentials at regular intervals, ideally at every instant. The key distribution problem between two devices (e.g., two ECG sensors [16]) has also attracted the attention of some researchers. In detail, in these solutions, each sensor derives the shared key from the acquired physiological signal, preventing the sensors from sharing any information beforehand [17,18]. In addition, the extraction of randomness from physiological signals has been recently scrutinised (e.g., ECG [19,20], EEG [21,22] and EMG [23]).

Regarding MBAN, the security of Implantable Medical Devices (IMDs) has attracted the attention of many researchers [24,25]. Even the FDA has alerted users of some vulnerabilities in commercial IMDs [26]. The proposed solutions to increase the security level of these critical devices are very diverse [27]. Some authors propose the usage of logs for auditing purposes [28] or the use of an external device that filters the messages sent to the implant [29]. The use of biometrics solutions, such as those based on fingerprints or iris, has been recently proposed [30,31]. Classical approaches based on symmetric [32,33], asymmetric [34,35] or hybrid ones [36] have been also suggested. Some authors have found interesting the combination of authentication schemes and distance bounding protocols [37]. Besides, some new research work focuses on the key distribution problem [16,38] and how to extract randomness from the signal acquired by the implant (mainly cardiac implants) [39–41].

The use of reliable Random Number Generators (RNGs) is crucial in security systems. Even well-known modern cryptographic solutions, such as the RSA private keys of HTTPS hosts, may have been compromised due to failures in the generation of nonces on networked devices [42]. When computational algorithms are used to generate random numbers, they are called Pseudorandom Number Generators (PRNGs). PRNGs depend on an initial value, called seed or key, and the outputted bitstream behaves as a random variable [43,44]. Alternatively, we can use physical phenomena with high entropy (e.g., atmospheric noise or decay of a radioactive source) as a source of randomness. This type of generators is called True Random Number Generators (TRNGs) [45,46].

In this article, we propose the design of a TRNG based on the GSR signal. As explained below, the parasympathetic nervous system controls the GSR signal. Therefore, instead of a physical phenomenon, we exploit a physiological signal that we carry with us—each user is the bearer of her or his random number generator. Besides, the GSR signal cannot be self-controlled, which prevents an attacker (or the carrier) from causing misbehaviour in the signal. As far as we know, Tuncer and Kaya [23] reported the only work close to our proposal that analyses the use of various biosignals, including the GSR signal, as a source for a random number generator. Unfortunately, in relation to the GSR signal, the proposal has been validated with only 12 subjects (much lower than 86 in our case; see Section 2.1) and the throughput (64 bits per second in the best case) is far from that offered by our proposal (1024 bits/s), as shown in the next sections.

1.2. Galvanic Skin Response

The electrical conductivity of our skin undergoes subtle alterations every time we are emotionally aroused. The Galvanic Skin Response (GSR)—also known as Electrodermal Activity (EDA) or Skin Conductivity (SC)—is often used, because of its sensitivity, to measure these variations. Therefore, the GSR measures the changes in the electrical characteristics of the skin. Humans have between two and five million sweat glands; men and women have the same number of glands, but male glands secrete five times more in size and volume [47]. Likewise, sweating is triggered when we are exposed to emotional stimulation. Perspiration through skin pores makes changes in the balance of positive

and negative ions in the secreted fluid. As a result, we can observe changes in skin conductance. Note that an increase in skin conductivity means a decrease in skin resistance.

The Autonomic Nervous System (ANS), which forms with the Somatic Nervous System (SNS) the Peripheral Nervous System (PNS), controls the functioning of many organs, muscles, and glands [48]. In detail, this regulation (proper behaviour of our body) is achieved by impulses from the brain (and/or spinal cord) and generated by autonomous neurons. Sweet glands are part of the glands mentioned above. In detail, sweating is driven and balanced by the ANS, and we cannot consciously control it. The ANS consists of the parasympathetic and the sympathetic nervous system [49]. The former is responsible for "rest and digest". Decreased heart rate, decreased sweating, or decreased blood pressure are some effects of its activation. The latter is responsible for the body's "fight or flight" reaction. That is, it helps to protect the body and is involved in functions such as pupils dilatation, increased heart rate or sweating [50]. Therefore, both systems are complementary to each other.

The recording of the GSR signal is non-invasive, and we only need two electrodes for its acquisition. Three are the most common placements: (1) index and middle fingers; (2) left and right side of palm; and (3) foot. In the market, we can find low-cost hardware platforms (e.g., BITalino or Libelium e-Health platform [51]) for the acquisition of biosignals. In Figure 1, we show an example of the electrode placement using the Bitalino platform for the signal acquisition. In detail, the exosomatic method with a small constant voltage is the most common approach to measure the GSR signal. The skin conductance (1/resistance) values are determined by measuring the changes in the current flow between the two electrodes, as the voltage is constant [52].

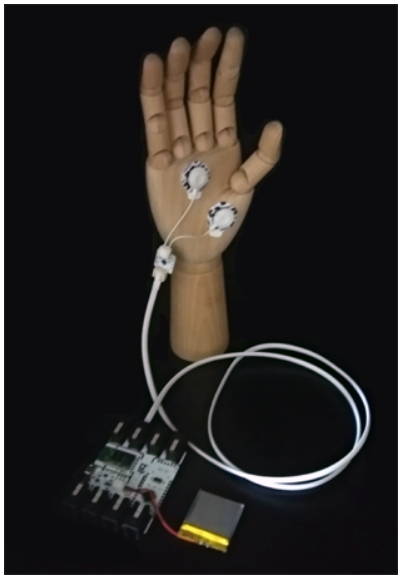

Figure 1. Electrodes placement for GSR acquisition.

2. Methods and Materials

2.1. Dataset Description

The randomness test batteries (e.g., DIEHARD [53] and NIST [54]) commonly used to verify the randomness quality of a random number generator require files of several tens of megabytes. For this reason, the GSR signals used in this study come from three well-known datasets:

1. The Affective Pacman (AffPac) dataset [55]. Twelve healthy users (aged 27 $\pm$ 3.9; 25% female) participated in the experiment. Several physiological signals were recorded simultaneously, including EEG, EOG and GSR signals.

2. DEAP dataset [56]. Thirty-two healthy participants (aged 28 ± 9; 50% female) volunteered for the experiment. The subjects watched several music videos while the physiological signals (e.g., EEG and GSR) were acquired.
3. AMIGOS dataset [57]. Forty healthy users participated in the experiment (aged 30.5 ± 9.5; 32.5% female). The participants watched short (16) and long (4) emotional videos. Three neuro-physiological signals (i.e., EEG, ECG and GSR signals) were recorded using wearable sensors. In our experiments, we discarded three files (subjects) because of their short length.

Note that we discarded the acquisition of our own GSR signals (e.g., using the Bitlanino platform) because, for our experimentation, we needed signals from many subjects and at the same time very extensive in time. As mentioned, in our experiments, we used signals from three datasets forming a total of 82 individuals (aged 28.5 ± 7.5; 35.8% female). Since no individuals present any severe pathology, we can then discard any bias in the output bits generated by the proposed TRNG. Furthermore, the signal acquisition process guaranteed that the GSR signals of the subjects in the dataset are statistically independent.

2.2. Methods

In our experiments, we focused exclusively on the GSR signal. We aimed to validate the hypothesis we can extract randomness from this vital signal. The proposed procedure is summarised in Algorithm 1 and explained below. First, for the GSR signal pre-processing, we followed a similar approach with all three datasets. As a first step, the data were down-sampled to 128 Hz. Then, a low-pass filter with 60 Hz cut-off frequency was applied. As an illustrative example, Figure 2 shows three minutes of a GSR signal.

Algorithm 1 GSR-TRNG.

procedure PRE-PROCESSING(GSR^{raw})
    `Down-sampling to 128 Hz`
    `Low-pass filter` ($[0 - 60Hz]$)

procedure GETENTROPY($GSR^{cleaned}$)
    `Split` $GSR^{cleaned}$ `into N-seconds` GSR-$windows$ `(N=4 in our experiments)`
    for `each GSR-window(`$x^{(j)}(t)$`)` **do**
        `Hilbert Transform:` $y^{(j)}(t) = h(t) * x^{(j)}(t)$
        `Entropy Extraction:` $g^{(j)}(t)_{(0,\ldots,7)} = uint8((uint32(abs(y^{(j)}(t) * 10^2))) >> 24)$

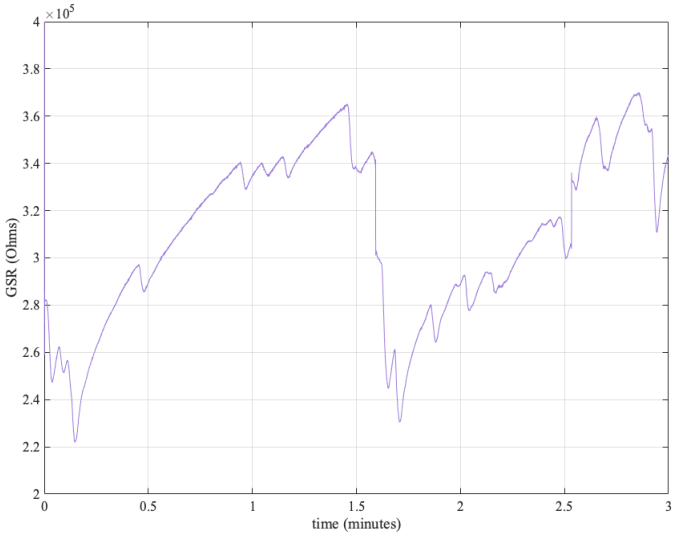

Figure 2. GSR signal.

After cleaning the GSR signal, we needed to extract randomness numbers from it. For this, we divided the GSR signal into windows of $N = 4$ seconds to be able to capture some variability in the signal—we fixed the size of the window by experimenting after analysing an extensive set of possible values. Secondly, we computed the Hilbert transform for each window. Hilbert transform can be interpreted as an all-pass filter in which all positive/negative frequencies are sifted $-90/90$ degrees, respectively. Mathematically, the Hilbert transform of a real, continuous-time signal is given by:

$$y(t) = h(t) * x(t) \tag{1}$$

where $h(t)$ represents the Hilbert transform kernel ($h(t) = \frac{1}{\pi t}, t \in (-\infty, \infty)$).

Finally, we extracted random bits from the Hilbert transform values. Mainly, we used an entropy extraction algorithm for this purpose. More precisely, using an accuracy of six decimal places, each value was converted to a 32-bit unsigned integer value, and then a byte was extracted from the Least Significant Bits (LSBs). It means that the proposed TRNG can generate $8 \times fs$ bits per second, with fs being the sampling rate used. The use of the LSBs is motivated by the fact that it is in these positions where there is more variability (randomness, formally stated) as confirmed by the results presented in the following sections. Mathematically, the extraction of random bits can be expressed as:

$$g(t)_{(0,\ldots,7)} = uint8((uint32(abs(y(t) * 10^2))) >> 24) \tag{2}$$

Once we specified the randomness extraction algorithm, we needed to assess the quality of the random numbers generated. For this purpose, we used the datasets introduced in Section 2.1. The reader can consult the following section for an in-depth security analysis of the proposed TRNG.

3. Results

We analysed the proposal from two perspectives. Firstly, the quality of the entropy source was studied, using the NIST SP 800-90B recommendation [58]. Secondly, the randomness of the random numbers generated was examined using well-known batteries of tests, such as DIEHARDER [53] and NIST [54].

3.1. Source Entropy Analysis

A cryptographic Random Bit Generator (RBG) is composed of three components: (1) an entropy source; (2) an algorithm responsible of storing and providing bits to the target application, and (3) the procedure for combining the two first components. In a nutshell, the entropy source model consists of an analogue noise source (in our case, the GSR signal, which is first cleaned with the procedure Pre-Processing in Algorithm 1) and a digitisation algorithm (procedure GetEntropy specified in Algorithm 1 and defined by Equations (1) and (2)).

For testing the entropy of RBGs, the NIST SP 800-90B recommendation proposes ten estimators, including the Markov and LZ78Y estimate among others for calculating the min-entropy [58]. The final estimation is the minimum value of all these tests. A file of 25 million 1s and 0s was generated using the third dataset to evaluate the entropy quality of the GSR signal. In most tests (see Table 1), the entropy value was close to the optimal (1) and even for the worst case remained very high (0.935). In this particular case, the t-tuple test sets the min-entropy value. This test evaluates the frequency of pairs, triples, and so on, and estimates the entropy per sample based on these frequencies [58]. From all the above, fortunately, we can conclude that the GSR signal together with the proposed digitisation algorithm seemed appropriate for cryptographic solutions.

Table 1. Min-entropy results (NIST SP 800-90B Suite).

Method	Min-Entropy
Most Common Value Estimate	0.99876
Collision Estimate	0.966577
Markov Estimate	0.999052
Compression Estimate	1
t-Tuple Estimate	0.935861
LRS Estimate	0.965143
MultiMCW Prediction Estimate:	0.999605
Lag Prediction Estimate	0.999152
MultiMMC Prediction Estimate	0.998977
LZ78Y Prediction Estimate	0.998780
Overall estimation	0.935861

In some occasions, the estimation of the entropy calculated on a very long sequence can produce an overestimation of the entropy—correlated sequences might be generated after a restart. If this is the case, the attacker could cause multiple restarts of the entropy source to generate an advantageous situation for her/him. The "restart" test is defined in the NIST SP 800-90B specification to evaluate this issue. As for generating data for this test, the GSR source was restarted 10^3 times, and then we recorded 10^3 consecutive values. In our case, we used the third dataset, in which the subjects were shown 20 different videos. Therefore, in our experiments, the reset of the physiological signal was simulated by exposing the subject to a different stimulation (video). Furthermore, to be even more confident, we repeated the test five times (i.e., from File-1 to File-5). As shown in Table 2, the five tests were passed successfully and confirmed that 0.94 was not an overestimate for the min-entropy.

Table 2. Restart tests (NIST SP 800-90B Suite).

File ID	Result
File-1	Pass
File-2	Pass
File-3	Pass
File-4	Pass
File-5	Pass
Final min-entropy estimation	**0.94**

3.2. *Randomness Analysis*

In Algorithm 1, we included an entropy distillation process (Procedure GetEntropy) to produce randomness. After the entropy analysis, we needed to assess the randomness quality of the bits generated by the GSR-TRNG. For a first visual inspection in Figure 3, we show an 8-bit grey scale image (512×512) of values generated by our TRNG. No anomalous patterns were detected, and the image behaves as the one generated by any other strong cryptographic random number generator. Several test batteries are commonly used (ENT [59], DIEHARDER [53] and NIST [54]) to analyse the randomness in depth. These tests require an input file of several hundred million bits. In our particular case, we generated a file of 30 MBytes by joining the GSR signals (signals of 84 subjects in total) of the three datasets introduced in Section 2.1.

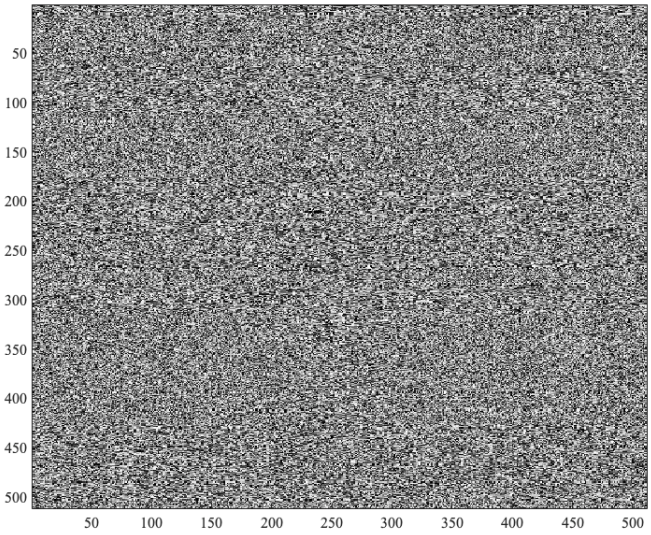

Figure 3. Random numbers generated by the proposed GSR-RNG.

ENT suite [59], which is not intended for cryptographic applications, is one of the test batteries usually used first to discard weak or faulty generators without the need for additional testing. Table 3 shows the results after analysing the 30 MByte file mentioned above. The entropy and compression results indicate that the file was extremely dense in terms of information (randomness). As for the chi-square test, which is very sensitive to detect weak generators, the results show no g suspicion of being not random. The arithmetic mean value confirmed that the proportion of ones and zeros were equal (i.e., there was no bias in the output). The serial correlation coefficient showed the high unpredictability of the bitstream—there was a low dependence between a particular bit and its predecessors.

Table 3. ENT results.

Entropy	7.999994
Optimum compression	0%
Chi square	235.33 (80.64%)
Arithmetic mean value	127.4990
Monte Carlo π value	3.143071846 (error 0.05%)
Serial correlation coefficient	−0.000129

To analyse whether there were no biases in the behaviour of each subject's signals, we performed an additional experiment by analysing them separately. Using the signal of the 37 subjects of the AMIGOS dataset, we generated a binary file of 800 KB for each of the subjects. Each of these files was analysed with the ENT suite. Figure 4 shows the result of the chi-square test. As shown in the figure, most values were within the optimal value (256) and $\pm$ the standard deviation. We can, therefore, conclude that the different subjects behaved similarly. In other words, there were no significant differences between the bitstreams generated from the different GSR signals corresponding to each subject.

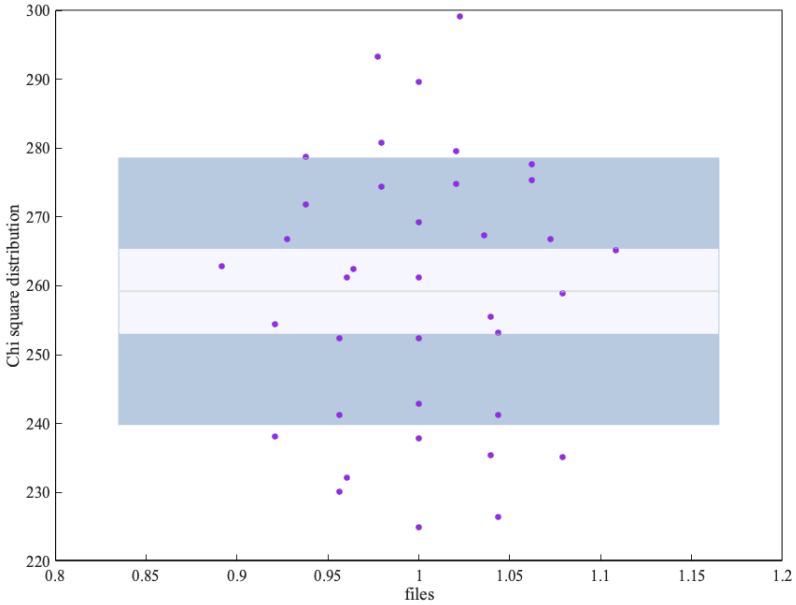

Figure 4. Bias analysis.

DIEHARDER [53] (a modern version of the Diehard battery), and NIST [54] are much more demanding test batteries than ENT. NIST has been designed to test RNGs that are devoted to cybersecurity solutions. DIEHARD consists of 15 test and the results obtained are summarised with a *p*-value in Table 4a. In detail, all tests were within the interval [0.025–0.975]—note that, due to a large number of *p*-values calculated, it would not be uncommon for some of them to be outside this range. Apart from being distributed within the interval mentioned above (0.05 of significance level), the critical point to consider the file under analysis random is that these *p*-values must follow a uniform distribution. We tested this hypothesis using a Kolgomorov–Smirnov test, which returned a decision that the *p*-values come from a uniform distribution at the 5% of the significance level. Therefore, we can conclude (95% of confidence) that there were no bad behaviours in the analysed bitstream (30 MByte file) and that all the DIEHARD tests were successfully passed. As mentioned above, NIST is often used in the context of cybersecurity and for formal verification of RNG designs. The NIST suite is made up of 15 tests, which examine bits, m-bit blocks or m-bit parts. Regarding the interpretation of the results, the first value corresponds with the *p*-value calculated for uniformity testing with the *p*-values obtained with a given test; the values in brackets represent the proportion of tests passing the corresponding test. The following equation gives the minimum number of tests (except for the random excursion test) that must be passed for each test:

$$mpr = (1 - \alpha) - 3 * sqrt(\frac{\alpha * (1 - \alpha)}{k}) \quad (3)$$

being $(1 - \alpha)$ the significance level and K the number of sequences tested. In our particular case, $\alpha = 0.01$ and $K = 100$, thus the minimum pass rate was 96. From the results in Table 4b, all the tests passed the uniformity test (*p*-values in the interval 0.01–0.99; $\alpha = 0.01$) and the proportion test was above the mentioned threshold ($mpr = 96$). Furthermore, the Kolgomorov–Smirnov confirmed the uniformity of all *p*-values (15 tests) with 1% of significance level. From all this, we can conclude that the bits generated by the TRNG based on GSR signals behaved as a random variable.

Table 4. DIEHARD and NIST Results.

(a) DIEHARD Results	
Birthdays	0.1079
OPERM5	0.1265
32x32 Binary Rank	0.5070
6x8 Binary Rank	0.6194
Bitstream	0.1318
OPSO	0.0386
OQSO	0.1792
DNA	0.1792
Count the 1s (stream)	0.9853
Count the 1s Test (byte)	0.2096
Parking Lot	0.0667
Minimum Distance (2d Circle)	0.5923
3d Sphere (Minimum Distance)	0.9626
Squeeze Test	0.8645
Sum Test	0.0340
Runs	0.2381 (up)
	0.6902 (down)
Craps	0.5847 (wins)
	0.3163 (throws)
(b) NIST Results	
Frequency	0.7792 (98/100)
Block Frequency	0.6787 (99/100)
Cumulative Sums	0.2974 (2/2)
	(99/100)
Runs	0.2368 (98/100)
Longest Run	0.7197 (100/100)
Rank	0.3345 (98/100)
FFT	0.8831 (99/100)
Non-Overlapping	0.5181 (148/149)
Template	(>99/100)
Overlapping Template	0.5749 (100/100)
Universal	0.3838 (99/100)
Approximate Entropy	0.0909 (100/100)
Random Excursions	0.6781 (8/8)
	(>61/62)
Random Excursions	0.5799 (18/18)
Variant	(>36/37)
Serial	0.8188 (2/2)
	(>99/100)
Linear Complexity	0.1296 (100/100)

As an additional experiment, we analysed whether there was any relationship between the random numbers generated by each user (GSR signal). If this were the case, it would be very advantageous for an attacker, since s/he could exploit the knowledge of a GSR signal (e.g., User-A) and predict the values of another signal (e.g., User-B). To assess this, using the 38 users of Dataset 3, we created a file of 800 KB. Next, we grouped the data of each file in words of different sizes ($m = \{8, 16, 32, 64\}$). For each of these word sizes, we computed the hamming distance between all the dataset pair combinations ($C_{38,2}$). We show the results obtained in Figure 5.

If there is no relation between the users (GSR signals), the calculated Hamming distance should follow a binomial distribution ($p(X = k) = \binom{m}{k} p^k (1-p)^{n-k}$; $E(X) = m * p$ and $\sigma^2 = n * p * (1-p)$) being m the size of the words and $p = 1/2$ as the zeros and the ones are equally likely). In our experiments (see Figure 5), as expected, the experimental values were almost identical to the theoretical

ones (i.e., a hamming distance of 4, 8, 16 and 32, respectively). Therefore, the advantage of an adversary of predicting the values of a user using the knowledge of other users' signals was zero.

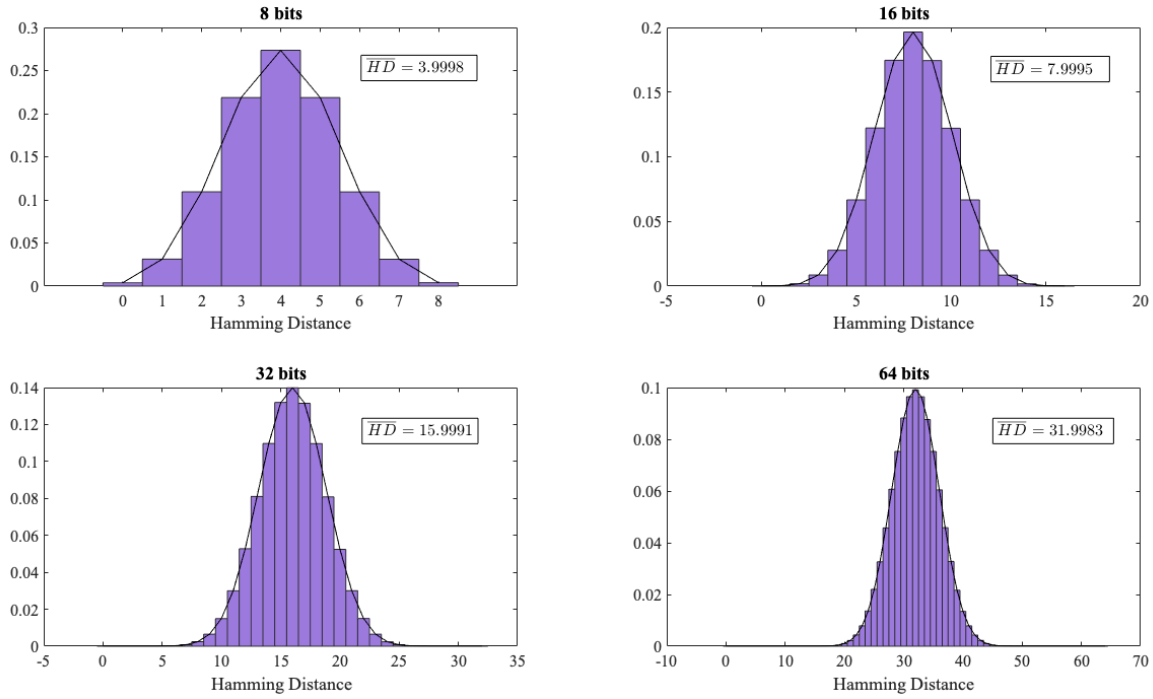

Figure 5. Hamming distance distribution.

Apart from the randomness tests, and as a final test, we analysed the TRNG as if it were used as a generator of a ciphering sequence (s) to encrypt a plain-text (m): $c = E(s, m) = s \oplus m$. In particular, using this approach, five different images (256×256 grayscale images), chosen randomly from the Internet, were used as inputs for the experiment. As for the ciphering sequence, bits were grouped in bytes and then regrouped into a matrix of the same size as the inputs images. As a first glance, Figure 6 shows the histogram of one of the tested images and its histogram after encryption. As expected, the encryption made the histogram uniform. Note that, if s (image with random values) follows a uniform distribution, and s and m are chosen independently of each other, the resulting value is uniformly distributed, since we combine them with the bitwise operation. This uniform distribution at the output makes it impossible for an attacker to extract any information from the original plaintext (image from the Internet in our example). Nowadays, NPCR and UACI tests are used to evaluate the strength of an image encryption technique against differential attacks [60]. In short, the first assesses the number of changing pixels and the second evaluates the changes in intensity, in both cases, between two encrypted images when the two plain images differ by one bit. In Table 5, we summarise the results of these test for the five examined images. Considering the thresholds given in [61], NPCR and UACI tests passed successfully at 0.05 significance level (i.e., $NPRC_{0.05} \geq 99.5693\%$ and $33.284\% \leq UACI_{0.005} \leq 33.6447\%$).

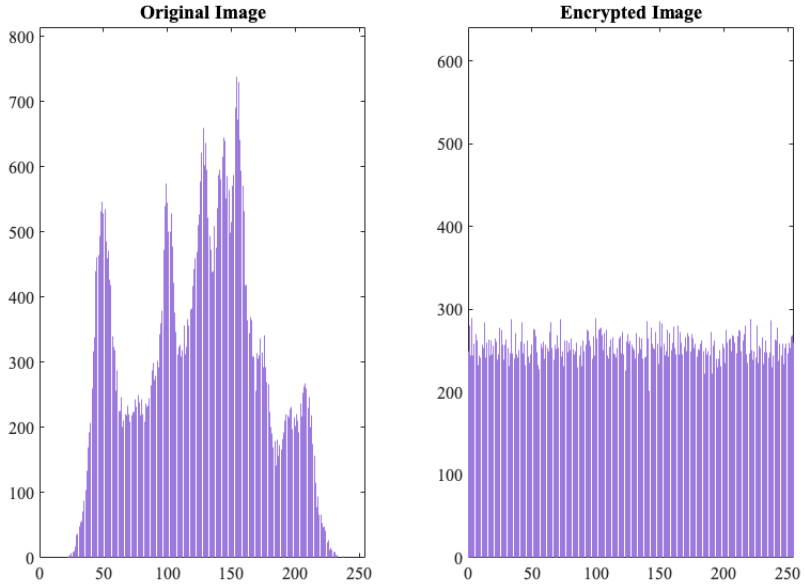

Figure 6. Original and encrypted statistical histograms.

Table 5. NPCR and UACI randomness tests.

	NPCR	**UACI**
File-1	99.6139%	33.6028%
File-2	99.6185%	33.6315%
File-3	99.5911%	33.2750%
File-4	99.6124%	33.4287%
File-5	99.6139%	33.4694%
Optimal value (256×256) [61]	$NPCR_{0.05} \geq 99.5693\%$ $NPCR_{0.01} \geq 99.5527\%$ $NPCR_{0.001} \geq 99.5341\%$	$33.2824\% \leq UACI_{0.005} \leq 33.6447\%$ $33.2255\% \leq UACI_{0.01} \leq 33.7016\%$ $33.1594\% \leq UACI_{0.001} \leq 33.7677\%$

4. Discussion and Conclusions

Today, there are many devices that monitor vital signs. These devices can be medical devices such as pacemakers or insulin pumps or general purpose devices such as sports watches or smart clothing with sensors. In any case, we have devices equipped with one or several sensors that transmit the acquired information (in most cases, wirelessly) to a central device. Although no one doubts the benefits of constant monitoring of our physiological parameters, access to these data only to authorised entities and their protection when transmitted through an insecure channel (mainly the radio channel) should be guaranteed from the design phase. Random number generators play a critical role in the design of cryptographic solutions for this purpose. Motivated by this fact, in this article, we have proposed a TRNG that benefits from a vital signal that is already being monitored by a sensor on the body. In particular, we have studied how to design a random number generator based on the GSR signal. Both the entropy source and the output randomness analysis confirm that the generated bitstreams behave as a random variable.

As shown in Algorithm 1, for the extraction of the randomness (Procedure GetEntropy), the Hilbert transform is used, which is usually used to construct the Analytic signal. Mathematically, given a signal $x(t)$ and its Hilbert transform $y(t)$, it is defined by $x_A(t) = x(t) + jy(t)$. In our particular case, we use only the imaginary part of the analytic signal that corresponds to the Hilbert transform itself. The reader may be tempted to think that the extraction of the entropy could be done from the signal itself (without any transformation). However, this was the first approach that we tested, and, although the output is entropic, a simple test such as the chi-square (ENT suite) clearly shows how the bits generated are non-random. Therefore, the use of Hilbert's transform is justified. Note that the

procedure for extracting random bits (see Equation (2) in Section 2.2) also plays a crucial role in our proposed TRNG.

In general terms, three elements are the main components of a TRNG: (1) noise source (GSR signal in our case); (2) digitisation algorithm; and (3) post-processing procedure (optional). In our case, we only have the first two elements since we consider that post-processing is not necessary. Among the most common post-processing techniques are bitwise XOR operations, Von Neumann algorithm or even the use of a hash function [62,63]. The use of these techniques is mandatory when the quality (randomness) of the output is not yet the desired. As shown in the in-depth analysis of the randomness (see Section 3.2), our generator successfully passes all the test batteries, and that is why our proposal dispenses with this stage.

A key parameter about any primitive cryptography is its performance. In the case of random number generators, high or moderately high throughput may be necessary for many applications. The proposed TRNG can generate 1024 bits per second (i.e., $8 \times fs = 8 \times 128$). This performance is far superior to that achieved by other random number generators using biosignals. In this context, the cardiac signal is the most studied physiological signal for this purpose. Solutions based on Interpulse Interval (IPI) values can generate between 2 and 14 bits per second [38,64], which is far below our performance. Even modern solutions based on the wavelet transform offer a through three times lower [19]. Concerning the GSR signal and the recently proposed TRNG [23], its throughput is 16 times lower at best than that of our approach. We can conclude from all this that our proposal offers excellent performance to be used in cybersecurity solutions.

As shown in this article, a new generation of TRNGs based on our vital signs can be designed. Apart from the GSR signal, and cardiac signals, other signals, such a the electrical activity of the brain (e.g., electroencephalogram) or the skeletal muscles (e.g., electromyogram) could be employed. Even for highly demanding applications, the combined use of various signals could give excellent results. As a conclusion, we can state that just as we still have much to learn from the human body within medicine, the use of the body is even less explored for cybersecurity tasks. In addition, it is worth mentioning that the use of sensors, integrated into a wide variety of devices, plays a critical role in the acquisition of the signal at stake.

Author Contributions: All authors contributed equally to this work in all tasks.

Funding: This work was supported by the Spanish Ministry of Economy and Competitiveness under the contract ESP-2015-68245-C4-1-P, by the MINECO grant TIN2016-79095-C2-2-R (SMOG-DEV), and by the Comunidad de Madrid (Spain) under the project CYNAMON (P2018/TCS-4566), co-financed by European Structural Funds (ESF and FEDER). This research was also supported by the Interdisciplinary Research Funds (HTC, United Arab Emirates) under the grant No. 103104.

Acknowledgments: The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, J.; Ma, Q.; Chan, A.H.; Man, S. Health monitoring through wearable technologies for older adults: Smart wearables acceptance model. *Appl. Ergon.* **2019**, *75*, 162–169. [CrossRef]
2. Wu, W.; Pirbhulal, S.; Sangaiah, A.K.; Mukhopadhyay, S.C.; Li, G. Optimization of signal quality over comfortability of textile electrodes for ECG monitoring in fog computing based medical applications. *Future Gener. Comput. Syst.* **2018**, *86*, 515–526. [CrossRef]
3. Wu, F.; Wu, T.; Yuce, M.R. An Internet-of-Things (IoT) Network System for Connected Safety and Health Monitoring Applications. *Sensors* **2018**, *19*, 21. [CrossRef] [PubMed]
4. Kompara, M.; Islam, S.H.; Hölbl, M. A robust and efficient mutual authentication and key agreement scheme with untraceability for WBANs. *Comput. Netw.* **2019**, *148*, 196–213. [CrossRef]

5. Yessad, N.; Bouchelaghem, S.; Ouada, F.S.; Omar, M. Secure and reliable patient body motion based authentication approach for medical body area networks. *Pervasive Mob. Comput.* **2017**, *42*, 351–370. [CrossRef]
6. Fortino, G.; Ghasemzadeh, H.; Gravina, R.; Liu, P.X.; Poon, C.C.Y.; Wang, Z. Advances in multi-sensor fusion for body sensor networks: Algorithms, architectures, and applications. *Inf. Fusion* **2019**, *45*, 150–152. [CrossRef]
7. Ambigavathi, M.; Sridharan, D. Energy efficient and load balanced priority queue algorithm for Wireless Body Area Network. *Future Gener. Comput. Syst.* **2018**, *88*, 586–593. [CrossRef]
8. Labati, R.D.; Muñoz, E.; Piuri, V.; Sassi, R.; Scotti, F. Deep-ECG: Convolutional Neural Networks for ECG biometric recognition. *Pattern Recognit. Lett.* **2018**. [CrossRef]
9. Ribeiro Pinto, J.; Cardoso, J.S.; Lourenço, A. Evolution, Current Challenges, and Future Possibilities in ECG Biometrics. *IEEE Access* **2018**, *6*, 34746–34776. [CrossRef]
10. Nakamura, T.; Goverdovsky, V.; Mandic, D.P. In-Ear EEG Biometrics for Feasible and Readily Collectable Real-World Person Authentication. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 648–661. [CrossRef]
11. Chan, H.L.; Kuo, P.C.; Cheng, C.Y.; Chen, Y.S. Challenges and Future Perspectives on Electroencephalogram-Based Biometrics in Person Recognition. *Front. Neuroinform.* **2018**, *12*, 66. [CrossRef]
12. Camara, C.; Peris-Lopez, P.; Gonzalez-Manzano, L.; Tapiador, J. Real-time electrocardiogram streams for continuous authentication. *Appl. Soft Comput.* **2018**, *68*, 784–794. [CrossRef]
13. Pinto, J.R.; Cardoso, J.S.; Lourenço, A.; Carreiras, C. Towards a Continuous Biometric System Based on ECG Signals Acquired on the Steering Wheel. *Sensors* **2017**, *17*, 2228. [CrossRef]
14. Wang, M.; Abbass, H.A.; Hu, J. Continuous authentication using EEG and face images for trusted autonomous systems. In Proceedings of the 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 12–14 December 2016; pp. 368–375. [CrossRef]
15. Wang, M.; Abbass, H.A.; Hu, J. EEG-based biometrics for person identification and continuous authentication. In *Information Security: Foundations, Technologies and Applications*; Security, Institution of Engineering and Technology: Stevenage, UK, 2018; pp. 311–346, doi:10.1049/PBSE001E_ch13.
16. Rostami, M.; Juels, A.; Koushanfar, F. Heart-to-heart (H2H): Authentication for Implanted Medical Devices. In Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS '13), Berlin, Germany, 4–8 November 2013; ACM: New York, NY, USA, 2013; pp. 1099–1112.
17. Venkatasubramanian, K.K.; Venkatasubramanian.; Banerjee, A.; Gupta, S.K.S. EKG-based key agreement in Body Sensor Networks. In Proceedings of the IEEE INFOCOM Workshops, Phoenix, AZ, USA, 13–18 April 2008; pp. 1–6.
18. Kim, J.; Cho, K.; Kim, Y.K.; Lim, K.S.; Shin, S.U. Study on peak misdetection recovery of key exchange protocol using heartbeat. *J. Supercomput.* **2018**. [CrossRef]
19. Camara, C.; Peris-Lopez, P.; Martín, H.; Aldalaien, M. ECG-RNG: A Random Number Generator Based on ECG Signals and Suitable for Securing Wireless Sensor Networks. *Sensors* **2018**, *18*, 2747. [CrossRef]
20. Chen, G. Are electroencephalogram (EEG) signals pseudo-random number generators? *J. Comput. Appl. Math.* **2014**, *268*, 1–4. [CrossRef]
21. Gavas, R.D.; Navalyal, G.U. Fast and secure random number generation using low-cost EEG and pseudo random number generator. In Proceedings of the International Conference on Smart Technologies For Smart Nation (SmartTechCon), Bengaluru, India, 17–19 August 2017; pp. 369–374.
22. Nguyen, D.; Tran, D.; Ma, W.; Nguyen, K. EEG-Based Random Number Generators. In Proceedings of the Network and System Security (NSS), Helsinki, Finland, 21–23 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 248–256.
23. Tuncer, S.A.; Kaya, T. True Random Number Generation from Bioelectrical and Physical Signals. *Comput. Math. Methods Med.* **2018**, *2018*, 3579575.
24. Marin, E.; Singelée, D.; Garcia, F.D.; Chothia, T.; Willems, R.; Preneel, B. On the (in)security of the Latest Generation Implantable Cardiac Defibrillators and How to Secure Them. In Proceedings of the Annual Conference on Computer Security Applications (ACSAC), Los Angeles, CA, USA, 5–9 December 2016; ACM: New York, NY, USA, 2016; pp. 226–236.
25. Slotwiner, D.J.; Deering, F.; Fu, K.; Russo, A.M.; Walsh, M.N.; Van Hare, G.F. Cybersecurity Vulnerabilities of Cardiac Implantable Electronic Devices. *Heart Rhythm* **2018**, *15*, e61–e67. [CrossRef]

26. Food and Drug Administration. FDA Warns Patients, Providers about Cybersecurity Concerns with Certain Medtronic Implantable Cardiac Devices. 2018. Available online: https://www.fda.gov/NewsEvents/Newsroom/FDAInBrief/ucm623191.htm (accessed on 26 April 2019).
27. Camara, C.; Peris-Lopez, P.; Tapiador, J.E. Security and privacy issues in implantable medical devices: A comprehensive survey. *J. Biomed. Inform.* **2015**, *55*, 272–289. [CrossRef]
28. Halperin, D.; Heydt-Benjamin, T.S.; Fu, K.; Kohno, T.; Maisel, W.H. Security and Privacy for Implantable Medical Devices. *IEEE Pervasive Comput.* **2008**, *7*, 30–39. [CrossRef]
29. Zhang, M.; Raghunathan, A.; Jha, N.K. MedMon: Securing Medical Devices Through Wireless Monitoring and Anomaly Detection. *IEEE Trans. Biomed. Circuits Syst.* **2013**, *7*, 871–881. [CrossRef]
30. Zheng, G.; Yang, W.; Valli, C.; Qiao, L.; Shankaran, R.; Orgun, M.A.; Mukhopadhyay, S.C. Finger-to-Heart(F2H): Authentication for Wireless Implantable Medical Devices. *IEEE J. Biomed. Health Inform.* **2018**. [CrossRef]
31. Hei, X.; Du, X. IMD Access Control During Emergencies. In *Security for Wireless Implantable Medical Devices*; Springer: New York, NY, USA, 2013; pp. 19–35, doi:10.1007/978-1-4614-7153-0_4.
32. Zheng, G.; Fang, G.; Shankaran, R.; Orgun, M.A. Encryption for Implantable Medical Devices Using Modified One-Time Pads. *IEEE Access* **2015**, *3*, 825–836. [CrossRef]
33. Srinivas, J.; Das, A.K.; Kumar, N.; Rodrigues, J. Cloud Centric Authentication for Wearable Healthcare Monitoring System. *IEEE Trans. Dependable Secur. Comput.* **2018**. [CrossRef]
34. Challa, S.; Das, K.; Odelu, V.; Kumar, N.; Kumari, S.; Khan, M.K.; Vasilakos, A.V. An efficient ECC-based provably secure three-factor user authentication and key agreement protocol for wireless healthcare sensor networks. *Comput. Electr. Eng.* **2018**, *69*, 534–554. [CrossRef]
35. Wazid, M.; Das, A.K.; Kumar, N.; Conti, M.; Vasilakos, A.V. A Novel Authentication and Key Agreement Scheme for Implantable Medical Devices Deployment. *IEEE J. Biomed. Health Inform.* **2018**, *22*, 1299–1309. [CrossRef]
36. Jang, C.S.; Lee, D.; Han, J.W.; Park, J.H. Hybrid security protocol for wireless body area networks. *Wirel. Commun. Mob. Comput.* **2011**, *11*, 277–288. [CrossRef]
37. Rasmussen, K.B.; Castelluccia, C.; Heydt-Benjamin, T.S.; Capkun, S. Proximity-based Access Control for Implantable Medical Devices. In Proceedings of the 16th ACM Conference on Computer and Communications Security, Chicago, IL, USA, 9–13 November 2009; ACM: New York, NY, USA, 2009; pp. 410–419. [CrossRef]
38. Pirbhulal, S.; Zhang, H.; Wu, W.; Mukhopadhyay, S.C.; Zhang, Y. Heartbeats Based Biometric Random Binary Sequences Generation to Secure Wireless Body Sensor Networks. *IEEE Trans. Biomed. Eng.* **2018**, *65*, 2751–2759. [CrossRef]
39. Seepers, R.M.; Strydis, C.; Sourdis, I.; Zeeuw, C.I.D. On Using a Von Neumann Extractor in Heart-Beat-Based Security. In Proceedings of the IEEE Trustcom/BigDataSE/ISPA, Helsinki, Finland, 20–22 August 2015; Volume 1, pp. 491–498. [CrossRef]
40. Ortiz-Martin, L.; Picazo-Sanchez, P.; Peris-Lopez, P.; Tapiador, J. Heartbeats Do Not Make Good Pseudo-Random Number Generators: An Analysis of the Randomness of Inter-Pulse Intervals. *Entropy* **2018**, *20*, 94. [CrossRef]
41. Chizari, H.; Lupu, E. Extracting Randomness from The Trend of IPI for Cryptographic Operators in Implantable Medical Devices. *arXiv* **2018**, arXiv:1806.10984.
42. Hastings, M.; Fried, J.; Heninger, N. Weak Keys Remain Widespread in Network Devices. In Proceedings of the 2016 Internet Measurement Conference (IMC '16), Santa Monica, CA, USA, 14–16 November 2016; ACM: New York, NY, USA, 2016; pp. 49–63. [CrossRef]
43. Garcia-Bosque, M.; Pérez-Resa, A.; Sánchez-Azqueta, C.; Aldea, C.; Celma, S. Chaos-Based Bitwise Dynamical Pseudorandom Number Generator On FPGA. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 291–293. [CrossRef]
44. Melià-Seguí, J.; Garcia-Alfaro, J.; Herrera-Joancomartí, J. J3Gen: A PRNG for Low-Cost Passive RFID. *Sensors* **2013**, *13*, 3816–3830. [CrossRef]
45. Abutaleb, M.M. A novel true random number generator based on QCA nanocomputing. *Nano Commun. Netw.* **2018**, *17*, 14–20. [CrossRef]

46. Grujić, M.; Rožić, V.; Yang, B.; Verbauwhede, I. A Closer Look at the Delay-Chain based TRNG. In Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 27–30 May 2018; pp. 1–5. [CrossRef]
47. Low, P.A. Chapter 51—Sweating. In *Primer on the Autonomic Nervous System*, 3rd ed.; Academic Press: Cambridge, MA, USA, 2012; pp. 249–251, doi:10.1016/B978-0-12-386525-0.00051-2.
48. Felten, D.L.; O'Banion, M.K.; Maida, M.S. 9—Peripheral Nervous System. In *Netter's Atlas of Neuroscience*, 3rd ed.; Elsevier: Amsterdam, The Netherlands, 2016; pp. 153–231, doi:10.1016/B978-0-323-26511-9.00009-6.
49. D., L.; Marrs, C. Chapter 2—The Autonomic Nervous System and Its Functions. In *Thiamine Deficiency Disease, Dysautonomia, and High Calorie Malnutrition*; Academic Press: Cambridge, MA, USA, 2017; pp. 27–57. [CrossRef]
50. Rea, P. Introduction to the Nervous System. In *Clinical Anatomy of the Cranial Nerves*; Rea, P., Ed.; Academic Press: Cambridge, MA, USA, 2014; doi:10.1016/B978-0-12-800898-0.00019-1.
51. Bayo-Monton, J.L.; Martinez-Millana, A.; Han, W.; Fernandez-Llatas, C.; Sun, Y.; Traver, V. Wearable Sensors Integrated with Internet of Things for Advancing eHealth Care. *Sensors* **2018**, *18*, 1851. [CrossRef]
52. Zangróniz, R.; Martínez-Rodrigo, A.; Pastor, J.M.; López, M.T.; Fernández-Caballero, A. Electrodermal Activity Sensor for Classification of Calm/Distress Condition. *Sensors* **2017**, *17*, 2324. [CrossRef]
53. Brown, R.G. Dieharder: A Random Number Test Suite v3.31.1. 2011. Available online: https://webhome.phy.duke.edu/~rgb/General/dieharder.php (accessed on 26 April 2019).
54. Bassham, L.E.; Rukhin, A.L.; Soto, J.; Nechvatal, J.R.; Smid, M.E.; Barker, E.B.; Leigh, S.D.; Levenson, M.; Vangel, M.; Banks, D.L.; et al. *SP 800-22 Rev. 1a. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*; Technical Report; 2010. Available online: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf (accessed on 26 April 2019).
55. Reuderink, B.; Poel, M.; Nijholt, A. The Impact of Loss of Control on Movement BCIs. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2011**, *19*, 628–637. [CrossRef]
56. Koelstra, S.; Muhl, C.; Soleymani, M.; Lee, J.; Yazdani, A.; Ebrahimi, T.; Pun, T.; Nijholt, A.; Patras, I. DEAP: A Database for Emotion Analysis ;Using Physiological Signals. *IEEE Trans. Affect. Comput.* **2012**, *3*, 18–31. [CrossRef]
57. Miranda Correa, J.A.; Abadi, M.K.; Sebe, N.; Patras, I. AMIGOS: A Dataset for Affect, Personality and Mood Research on Individuals and Groups. *IEEE Trans. Affect. Comput.* **2018**. [CrossRef]
58. Turan, M.S.; Barker, E.; Kelsey, J.; McKay, K.; Baish, M.; Boyle, M. *NIST Special Publication 800-90B. Recommendation for the Entropy Sources Used for Random Bit Generation*, 2018. Available online: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf (accessed on 26 April 2019).
59. Walker, J. Randomness Battery. 1998. Available online: http://www.fourmilab.ch/random/ (accessed on 26 April 2019).
60. Özkaynak, F. Role of NPCR and UACI tests in security problems of chaos based image encryption algorithms and possible solution proposals. In Proceedings of the International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey, 5–7 October 2017; pp. 621–624. [CrossRef]
61. Wu, Y.; Noonan, J.P.; Agaian, S. NPCR and UACI Randomness Tests for Image Encryption. *Cyber J. Multidiscip. J. Sci. Technol.*, **2011**, 31–38. Available online: https://pdfs.semanticscholar.org/2b47/9abce221135af6065f9f8352e09cbfb5733a.pdf (accessed on 26 April 2019).
62. Rožić, V.; Yang, B.; Dehaene, W.; Verbauwhede, I. Iterating Von Neumann's post-processing under hardware constraints. In Proceedings of the IEEE International Symposium on Hardware Oriented Security and Trust (HOST), McLean, VA, USA, 3–5 May 2016; pp. 37–42.
63. Li, C.; Wang, Q.; Jiang, J.; Guan, N. A metastability-based true random number generator on FPGA. In Proceedings of the IEEE 12th International Conference on ASIC (ASICON), Guiyang, China, 25–28 October 2017; pp. 738–741.
64. Altop, D.K.; Levi, A.; Tuzcu, V. Deriving cryptographic keys from physiological signals. *Pervasive Mob. Comput.* **2017**, *39*, 65–79. [CrossRef]