entropy

MDPI

Article

Information Theory Based Evaluation of the RC4 Stream Cipher Outputs

Evaristo José Madarro-Capó [1], Carlos Miguel Legón-Pérez [1], Omar Rojas [2,*] and Guillermo Sosa-Gómez [2]

[1] Facultad de Matemática y Computación, Instituto de Criptografía, Universidad de la Habana, Habana 10400, Cuba; ejmcapo@gmail.com (E.J.M.-C.); clegon58@gmail.com (C.M.L.-P.)
[2] Facultad de Ciencias Económicas y Empresariales, Universidad Panamericana, Álvaro del Portillo 49, Zapopan 45010, Jalisco, Mexico; gsosag@up.edu.mx
* Correspondence: orojas@up.edu.mx; Tel.: +52-331-368-2200

Abstract: This paper presents a criterion, based on information theory, to measure the amount of average information provided by the sequences of outputs of the RC4 on the internal state. The test statistic used is the sum of the maximum plausible estimates of the entropies $H(j_t|z_t)$, corresponding to the probability distributions $P(j_t|z_t)$ of the sequences of random variables $(j_t)_{t\in T}$ and $(z_t)_{t\in T}$, independent, but not identically distributed, where z_t are the known values of the outputs, while j_t is one of the unknown elements of the internal state of the RC4. It is experimentally demonstrated that the test statistic allows for determining the most vulnerable RC4 outputs, and it is proposed to be used as a vulnerability metric for each RC4 output sequence concerning the iterative probabilistic attack.

Keywords: RC4; iterative probabilistic attacks; entropy; randomness

Citation: Madarro-Capó, E.J.; Legón-Pérez, C.M.; Rojas, O.; Sosa-Gómez, G. Information Theory Based Evaluation of the RC4 Stream Cipher Outputs. *Entropy* **2021**, *23*, 896. https://doi.org/10.3390/e23070896

Academic Editors: Nicusor Minculete and Shigeru Furuichi

Received: 22 April 2021
Accepted: 12 July 2021
Published: 14 July 2021

1. Introduction

In [1], the iterative probabilistic attack was proposed to reconstruct the internal state of the RC4 algorithm, starting from knowing an output sequence, which was successively improved in [2,3]. In essence, these attacks attempt to extract information about the content of the internal state $\{(j_t, S_t) : t = 1, \ldots, T\}$ of the algorithm RC4 stream cipher from a known output sequence $\{(z_t) : t = 1, \ldots, T\}$. For this, the conditional probabilities $P(j_t|z_t)$ and $P(S_t|z_t)$ are iteratively recalculated. This type of attack does not yet violate RC4, but it constitutes a serious potential threat to its security, which should not be ignored. Concerning this threat, a criterion has been developed to assess the vulnerability of an RC4 output to this type of attack. The test statistic used is based on the entropy of the conditional probability distributions $P(j_t|z_t)$ for the z_t that appear in the evaluated sample. The test statistic was proposed considering that the values and position of these z_t determine their probability distribution and associated entropy. The lower the value of the statistic, the more vulnerable the evaluated sample is the lower the attacker's uncertainty will be about the value of the variable j_t.

This result can have various applications, since it allows for an evaluation of a set of RC4 output sequences according to their vulnerability or theoretical strength in face of iterative probabilistic attacks. This criterion can characterize the keys that cause the greatest vulnerability, which can lead to the identification of a new class of weak keys. In this work, experimental results evaluating the RC4 output sequences, according to their vulnerability to probabilistic attacks, are presented.

The structure of this work is as follows: in Section 2 the basic concepts of the research topic are described, which includes the description of the RC4 algorithm and the reports associated with the iterative probabilistic attack; Section 3 introduces the statistic used to evaluate the vulnerability of the RC4 outputs concerning the iterative probabilistic attack; Section 4 details the pre-calculation of frequencies that allow the estimation of

the joint, marginal, and conditional probabilities and, in turn, the estimation of the entropies that will be used for the statistical calculation on the output sequences of RC4; in Section 5 experiments are performed to validate the proposed statistician. The results of applying the statistic on RC4 output sequences are illustrated; finally, Section 6 presents some conclusions.

2. Preliminaries

2.1. Description of the RC4 Stream Encryption Algorithm

The RC4 algorithm [4] stands out from other stream ciphers for its wide use in different applications and protocols. The RC4 stream cipher [4] is optimized to be used in 8-bit processors, being extremely fast and exceptionally simple. It was included in network protocols, such as secure sockets layer (SSL), transport layer security (TLS), wired equivalent privacy (WEP), Wi-Fi protected access (WPA), and in various applications used in Microsoft Windows, Lotus Notes, Apple Open Collaboration Environment (AOCE), and Oracle Secure SQL [4]. In the last decade, some applications [5,6] have avoided RC4 encryption, given some found weaknesses [7]. However, although it is not considered very secure [8], RC4 continues to motivate research nowadays [8–10]. Furthermore, this cipher is a good option to measure the effectiveness of methods that analyze weaknesses in stream ciphers related to those already known in RC4 [11–14], or to check the performance of hardware or software schemes that make use of cryptography [15–17].

The RC4 has two main components, the key scheduling, and the pseudorandom number generator. The key scheduling generates an internal random permutation S of values from 0 to 255, from an initial permutation, a (random) key K of l-byte length, and two pointers i and j. The maximal key length is of $l = 256$ bytes, see Algorithm 1.

Algorithm 1 RC4 key-scheduling.

for $i = 0 \rightarrow 255$ **do**
    $S[i] \leftarrow i$
end for
$j \leftarrow 0$
for $i = 0 \rightarrow 255$ **do**
    $j \leftarrow (j + S[i] + K[i \mod l]) \mod n$
    Swap $S[i]$ and $S[j]$
end for

The main part of the algorithm is the pseudo-random number generator that produces one-byte output at each step. As usual, for stream ciphers, the encryption will be a XOR of the pseudo-random sequence with the message, see Algorithm 2.

Algorithm 2 RC4 pseudo-random generator.

$i \leftarrow 0$
$j \leftarrow 0$
while Generating Output **do**
    $i \leftarrow (i + 1) \mod 256$
    $j \leftarrow (j + S[i]) \mod 256$
    Swap $S[i]$ and $S[j]$
    Output $S[(S[i] + S[j]) \mod 256]$
end while

For the RC4 stream cipher, several modifications have been proposed, while some modified only certain components or some operations, others completely changed the algorithm, see [18]. It is important to note that even RC4 variants have had a lot of attention in the scientific community, see [19].

The RC4 stream cipher, in its definition, does not distinguish the use of IV initialization vectors [4]. However, it is well known that in practical applications of RC4, as in many other stream ciphers, an IV initialization vector with a secret key is used to form a session key. The proposed method is independent of the approach used; it simply works on the final input used as input to the cipher.

2.2. Iterative Probabilistic Attacks

Here we discuss three important results on probabilistic attacks that try to reconstruct the internal state of RC4 from knowing an output sequence $\{(z_t) : t = 1, \ldots, T\}$. In [1], the central idea, proposed by Knudsen et al., was to conveniently use Bayes's Theorem to recalculate the probabilities $P(j_t|z_t)$ and $P(S_t|z_t)$, for each $t \in T$. In essence, they worked on obtaining probabilistic information about the two variables j_t, S_t from z_t. It was reported that a low probability of success and a high volume of work were achieved. To be successful, it was required to know the values of at least d elements of S_0, with $d \in \{150, \ldots, 160\}$. The results presented in [1] are independent of the key scheduling and the key size. For sequences of length $T = 256 = 2^8$, the volume of work was 2^{48} in each iteration. In [2], Knudsen's method was improved by reducing the number of elements of the permutation that must be known and maintains the same workload of 2^{48} in each iteration. The essential difference is that a more exact way of recalculating the probabilities was proposed using the entire Z output sequence instead of just the z_t value to increase the probability of success. Experiments were reported for RC4 with $n = 3$ and $n = 4$. Finally, in [3], Golic and Morgari used the same probabilities of the previous article; the novelty in that work was that it proposed a set of 7 improvements to the probabilistic algorithm itself and estimated the minimum number d of elements in S_0 that must be known a priori so that the attack recovers the correct S_0 permutation, concluding that $d \in \{26, \ldots, 85\}$, which is a substantial improvement compared to $d \in \{150, \ldots, 160\}$. The workload remained at 2^{48} probabilities that must be calculated at each iteration.

In summary, in the three aforementioned articles, it was reported that these attacks have a low probability of success when no element of the permutation is known a priori, which is why it is concluded that they are not currently applicable to real RC4. In such articles, the authors model the ignorance over the internal state assuming the initial uniform probability distribution for S and j. It is essential to note that increasing the precision of the recalculated probabilities reduced the number d of elements of the permutation that must be known a priori. Knudsen et al. got $d \in \{150, \ldots, 160\}$, while Golic and Morgari reduced it to $d \in \{26, \ldots, 85\}$. The previous result suggests that by increasing the precision of the calculated probabilities in different ways or by improving the iterative algorithm, it could be possible to achieve $d \approx 0$, i.e., to recover the complete permutation without knowing any of its elements a priori, which constitutes a serious threat to the safety of the RC4.

2.3. Entropy As a Measure of Uncertainty

Let X be a discrete random variable with possible values x_i and respective probabilities $p_i = P(x_i)$, with $i = 1, \ldots, k$. Then, Shannon's discrete entropy function $H(p_1, \ldots, p_k)$ [20] is defined as

$$H(X) = -\sum_{i=1}^{k} p_i \log p_i. \tag{1}$$

When $p_i = 1/k$ for all $i = 1, \ldots, k$, the maximum uncertainty about the value of X is obtained, so the entropy reaches its maximum value, equal to

$$H_{max} = H\left(\frac{1}{k}, \ldots, \frac{1}{k}\right) = log(k). \tag{2}$$

When there is an i' such that $p_{i'} = 1$ and $p_i = 0$ for all $i \neq i'$, there is no uncertainty about the value of X, so the entropy reaches its minimum value, equal to $H_{min} = 0$.

3. Definition of the Proposed Test Statistic

In this work, the information that z_t contributes on j_t probabilistically, by means of a non-uniform probability distribution, will be modeled from the knowledge of z_t. To support this proposal, we start from the relationship between z_t and j_t, and the result of [4] on the non-equi-probability of the permutation S at the beginning of the pseudo random generation algorithm (PRGA) stage.

Solving for j_t in the equation that defines z_t in the RC4 algorithm, we obtain:

$$j_t = S_t^{-1}\Big[S_t^{-1}[z_t] - S_t^{-1}[i_t]\Big]. \tag{3}$$

For $t = 0$, we have $j_0 = S_0^{-1}[S_t^{-1}[z_0] - S_0^{-1}[i_0]]$. Note that the values of i_0 and z_0 are known, while S_0 is unknown, therefore, the distribution of j_0 is determined by S_0. Taking into account that in [4] it is shown that S_0 does not follow a uniform distribution, it is considered that for $t = 0$, when z_0 is known, this property of non-uniformity is translated to j_0.

Expression (3) does not allow the calculation of j_t since S_t is unknown. However, it allows theoretically arguing the non-equi-probability of j_t, conditional on knowing the value of z_t (due to the non-equi-probability of S).

Denoting by t^* the smallest value of t, such that for $t > t^*$ it is true that S_t follows a uniform distribution. In [21,22], the authors tried to estimate the value of t^*. From this definition, and following the same reasoning as for S_t, at $t = 0$, the beginning of the first iteration; it can be assumed that for $t \in \{0, \ldots, t^* - 1\}$ the conditional probability distribution $P(j_t|z_t)$, is non-uniform. In [22], it is described that it is possible to find biases in the output bytes of RC4 up to $t^* = 512$. Thus, for $t \in \{0, \ldots, 511\}$ the conditional probabilities $P(j_t|z_t)$ do not fit a uniform distribution.

3.1. Basis of the Evaluation Criterion

The criterion will be limited to considering only the variable j_t and its conditional probabilities $P(j_t|z_t)$. This was taken into account because the knowledge without errors of the sequence $\{(j_t) : t = 1, \ldots, T\}$ allows to reconstruct S_0 [5]. The central idea of the criterion is based on the different values z_t that appear at each time step $t \in T$. This can cause different initial conditional probability distributions $P(j_t|z_t)$ to appear at each $t \in T$. This is the essence of the proposed test statistic; i.e., it will take into account which values z_t appear in the sample and in which places (times t) each one of those values z_t appears. Under this condition, two samples with different frequency distributions z_t will have different vulnerabilities to attack. Even between two samples with the same frequency distribution of the values z_t, the effectiveness would vary depending on their places of appearance.

3.2. Definition of the Test Statistic

To measure these differences, a 256×256 matrix was pre-calculated at each time $t = \{1, \ldots, T\}$, in which the columns represent all the possible values of $z_t = 0, \ldots, 255$ and the rows each value of $j_t = 0, \ldots, 255$. The element (j, z) of the matrix, corresponding to row j and column z, constitutes the conditional probability $P(j_t|z_t)$, at time $t \in T$. In this way, each column will be the distribution of conditional probabilities $P(j_t|z_t)$, which probabilistically represents the information about j that causes the appearance of z at this time. The most interesting question one might ask is: how can one compare two different columns? For example, how can one compare two probability distributions? More exactly, which distribution is associated with the greatest uncertainty about j? To solve this problem, the concept of Shannon's discrete entropy will be used.

For each column, the entropy will be calculated, denoted by H_z^t, which is a direct measure of the uncertainty about j_t, when in the place t of the sample appears the z_t value associated with that column. It is important to mention that the entropy value characterizes the distribution of the 256 possible values of j_t in a single value, facilitating the comparison between probability distributions. By entropy's properties, it is satisfied that if $H_z^t = 0$, then the value of the j_t variable is determined by z_t, while if $H_z^t = 8$, knowledge of z_t does not provide any information about the value of j_t.

To evaluate in a sample of length T, the total uncertainty over j, the entropy associated with the value z_t that appeared at each time $t = 1, \ldots, T$ will be added over all times. Then, the expression of the test statistic will be:

$$Q = \sum_{t=0}^{T} H_z^t, \tag{4}$$

where

$$H_z^t = -\sum_{j_t=0}^{255} P(j_t|z_t) log P(j_t|z_t). \tag{5}$$

The expected value $\mu = E(Q)$ and the variance $\sigma^2 = V(Q)$ of the statistic Q, are expressed from the expected value and the variance of the conditional entropies H_z^t in the T times and are given by

$$\mu = E(Q) = \sum_{t=0}^{T} E(H_z^t), \tag{6}$$

and assuming that the H_z^t, with $t = 1, \ldots, T$, are independent of each other,

$$\sigma^2 = V(Q) = \sum_{t=0}^{T} V(H_z^t). \tag{7}$$

For each entropy H_z^t that appears as an addition in the expression of Q, its distribution can be approximated by a normal distribution according to the result of [23]. However, this plug-in estimator is known to be biased. Its bias and variance [24,25], are given by

$$E(\widehat{H} - H) = -\frac{k-1}{2n} + \frac{1}{12n^2}\left(1 - \sum_{i=1}^{k} \frac{1}{p_i}\right) + O(n^{-3}), \tag{8}$$

and

$$Var(\widehat{H}) = \frac{1}{n}\left(\sum_{i=1}^{k} p_i \, ln^2(p_i) - H^2\right) + \frac{k-1}{2n^2} + O(n^{-3}), \tag{9}$$

where n is the sample size. If the bias's expression terms that include unknown parameters are depreciated, then the bias is calculable when the cardinality of the alphabet is known, as in this case, but the variance is not since it depends on the unknown probabilities p_i. In this work, the point estimation of the mean $\mu = E(Q)$ and the variance $\sigma^2 = V(Q)$ of the Q statistic was carried out directly, using the expressions

$$\widehat{\mu} = \widehat{E(Q)} = \sum_{t=0}^{T} \widehat{E(H_z^t)}, \tag{10}$$

and

$$\widehat{\sigma^2} = \widehat{V(Q)} = \sum_{t=0}^{T} \widehat{V(H_z^t)} \tag{11}$$

respectively, based on the point estimation [26] of the means $\widehat{E(H_z^t)}$ and the variances $\widehat{V(H_z^t)}$ of each entropy H_z^t for each time t, with $t = 1, \ldots, T$.

The lower the value of the Q-statistic, the less uncertainty about j, and, therefore, the sample would be more vulnerable to these attacks. To evaluate a set of samples of equal length, it is enough to calculate the test statistic for them and sort them increasingly. To compare samples of different lengths, the statistics obtained can be divided between the lengths of their respective samples, obtaining the average uncertainty per symbol and comparing in the same way.

3.3. Decision Criteria Using the Q-Statistic

The Q-statistic is defined as the sum of T random variables H_z^t. Following the results obtained in [23] by Zhang and Zhang, the plug-in entropy estimator, used to estimate the entropy considered in this work, follows an approximately normal distribution. In this way, assuming independence between the random variables H_z^t, the Q-statistic follows a normal distribution $N(\mu, \sigma^2)$ with mean μ and variance σ^2 because the sum of Normal independent variables is also normal. Then, it is possible to approximate the distribution of the Q-statistic to a random variable with standard normal distribution.

$$Q_{01} = \frac{Q - \mu}{\sigma} \sim N(0, 1), \tag{12}$$

where σ is the standard deviation of Q.

As mentioned above, the permutation of RC4 has biases that are transferred to j and the output z. The appearance of biases in the distribution $P(j_t|z_t)$ provides alterations in the values of H_z^t and consequently in the distribution of the Q-statistic, leading to the appearance of extreme values. The appearance of these extreme values adds to the distribution of Q a slight asymmetry on left tail since the alteration in the distribution of $P(j_t|z_t)$ decreases the value of H_z^t and, therefore, $Q \ll \mu$. For this reason, we will work with the standard normal distribution $N(0, 1)$ with a single tail, in this case with a left tail, and using a significance level α, it is concluded that the sequences from RC4 that provides more information about the variable j of the internal state are those with the lowest value of Q, such that $Q_{01} < Z_\alpha$.

4. Pre-Computing of Probabilities and Estimation of Entropies

The proposed method is divided into two phases, following an idea similar to a time memory trade off (TMTO) attack [27]. The first is the precomputation phase, often called the offline phase, where the probabilities and entropy in each time of T are estimated over each output value z_t. The objective of this phase is to estimate the information, in general, that provides the output occurrence z_t on the variable j_t. This phase is executed only once, and then used repeatedly in the next phase for the evaluation of N outputs of the RC4. Although the second is referred to as the real-time or online phase, where it captures a sample of RC4 keystream and checks if this happens to be in the tables below. Each of the M outputs were generated from initializing the RC4 with M random inputs of 20 bytes each.

To estimate the conditional probabilities $P(j_t|z_t)$, at each time $t \in T$ for all possible values z_t, a pre-calculation of frequencies was performed, and thus the entropies were estimated.

To make a good estimation of the probabilities, in this work, we used M = 262,144,000 outputs of RC4 to reliably obtain as many biases as possible that RC4 has and taking into account the size $k = 256$ of the alphabet.

4.1. Frequency Pre-Calculation

To calculate the frequencies, M = 262,144,000 outputs of the RC4 of length $T = 512$ were generated and, at each time $t \in T$, the value of the pair (j_t, z_t) was checked, obtaining for each fixed z_t the joint distribution (j_t, z_t) varying j_t. The value of M was chosen in order to obtain an expected frequency of

$$E(f_{(j_t,\, z_t)}) = 262{,}144{,}000/(256 \times 256) = 4000 \tag{13}$$

observations, by category, under the hypothesis of equi-probability.

A matrix of 256×256 was obtained for each time $t = 1, \ldots, 512$ which represents each value of $z_t = 0, \ldots, 255$ per column and in the rows each value from $j_t = 0, \ldots, 255$. Thus, we have in row j, column z, the frequency $f_{(j_t,\ z_t)}$ of joint appearance of the pair (j_t, z_t) at time t (see Table 1).

Table 1. Frequency of joint appearance.

J/Z	**0**	**1**	**…**	**255**	
0	f(0,0)	f(0,1)	…	f(0,255)	$f_{(j_t=0)}$
1	f(1,0)	f(1,1)	…	f(1,255)	$f_{(j_t=1)}$
⋮	⋮	⋮	…	⋮	⋮
255	f(255,0)	f(255,1)	…	f(255,255)	$f_{(j_t=255)}$
	$f_{(z_t=0)}$	$f_{(z_t=1)}$	⋯	$f_{(z_t=255)}$	M

4.2. Estimation of Joint, Marginal, and Conditional Probabilities

From the joint frequencies $f_{(j_t,\ z_t)}$ we can obtain the marginal frequency $f_{(\ z_t)}$ at each time t, to estimate the joint probability $P(j_t, z_t)$ and the marginal probability $P(z_t)$, in order to reach an estimate of the conditional probability $P(j_t|z_t)$, through the Bayes formula

$$\widehat{P}(j_t, z_t) = \frac{f_{(j_t,\ z_t)}}{M}, \tag{14}$$

$$\widehat{P}(z_t) = \frac{f_{(\ z_t)}}{M} = \frac{\sum_{j=0}^{255} f_{(j_t,\ z_t)}}{M}, \tag{15}$$

and thus

$$\widehat{P}(j_t|z_t) = \frac{\widehat{P}(j_t,\ z_t)}{\widehat{P}(z_t)} = \frac{\frac{f_{(j_t,\ z_t)}}{M}}{\frac{\sum_{j=0}^{255} f_{(j_t,\ z_t)}}{M}} = \frac{f_{(j_t,\ z_t)}}{\sum_{j=0}^{255} f_{(j_t,\ z_t)}}. \tag{16}$$

From the estimation of these probabilities, a table like the Table 2 is obtained, which now contains the conditional probability $\widehat{P}(j_t|z_t)$ for each time $t = 1, \ldots, 512$.

Table 2. Probabilities of j conditional on z at each time t.

J/Z	**0**	**…**	z_t	**…**	**255**
0	⋮				
⋮			⋮		⋮
j_t			$\widehat{P}(j_t/z_t)$		
⋮	⋮		⋮	…	⋮
255					

4.3. Entropy Estimation

For each time $t \in T$, the entropy $H_z^t = -\sum_{j_t=0}^{255} P(j_t|z_t) log P(j_t|z_t)$ was estimated, using the plug-in estimator [28]. This constitutes the entropy of the distribution of the j conditioned to the value z_t of that column. Thus, at each time $t \in T$, 256 values of $\widehat{H}_z^t$ are obtained. The output z_t with the highest entropy $\widehat{H}_z^t$ (tighter distribution to the uniform) provides the less information on j. Uniting the results obtained for the $T = 512$ times, a matrix of 256×512 is obtained which contains per column each value of $t = 1, \ldots, 512$ and in the rows each value of $z = 0, \ldots, 255$. In each category (z, t) will be the entropy value $\widehat{H}_z^t$ corresponding to row z and column t (see Table 3).

Table 3. Entropy of the j distributions conditional on z at each time.

Z/T	1	2	...	512
0	$\widehat{H}_0^1$	$\widehat{H}_0^2$	...	$\widehat{H}_0^{512}$
1	$\widehat{H}_1^1$	$\widehat{H}_1^2$	...	$\widehat{H}_1^{512}$
⋮	⋮	⋮	...	⋮
255	$\widehat{H}_{255}^1$	$\widehat{H}_{255}^2$	...	$\widehat{H}_{255}^{512}$

Then, to evaluate a particular sample, the value $\widehat{H}_z^t$ corresponding to the place (t, z_t) of the matrix is added using the statistic at each time t. In this way, a random variable of the type is obtained at each time

$$\begin{pmatrix} \widehat{H}_{z=0}^t & \dots & \widehat{H}_{z=255}^t \\ P(z_t = 0) & \dots & P(z_t = 255) \end{pmatrix}, \tag{17}$$

whose expected value will be:

$$E[\widehat{H}_z^t] = \sum_{i=0}^{255} \widehat{P}(z_t = i) \cdot \widehat{H}_z^t = \widehat{H}^t(J/Z), \tag{18}$$

which constitutes the average uncertainty over j, at time t, when z_t is known, i.e., the conditional entropy.

5. Experimental Evaluation

In the experiments ran for the present article, $T = 512$ times will be taken, as in the pre-calculation stage and N = 10,000 output sequences of the RC4 were generated from N random entries of 20 bytes each. The T value can be a variable parameter depending on the size required for the sample given the pre-calculation performed. For higher value selection of this parameter, it is necessary to deepen the theoretical comparison between the times and carry out more experiments. Figure 1 shows the distribution of the Q-statistic calculated at the 10,000 sequences generated. The left skewness illustrates the appearance of biases in the $P(j_t|z_t)$ distribution that decrease the value of Q.

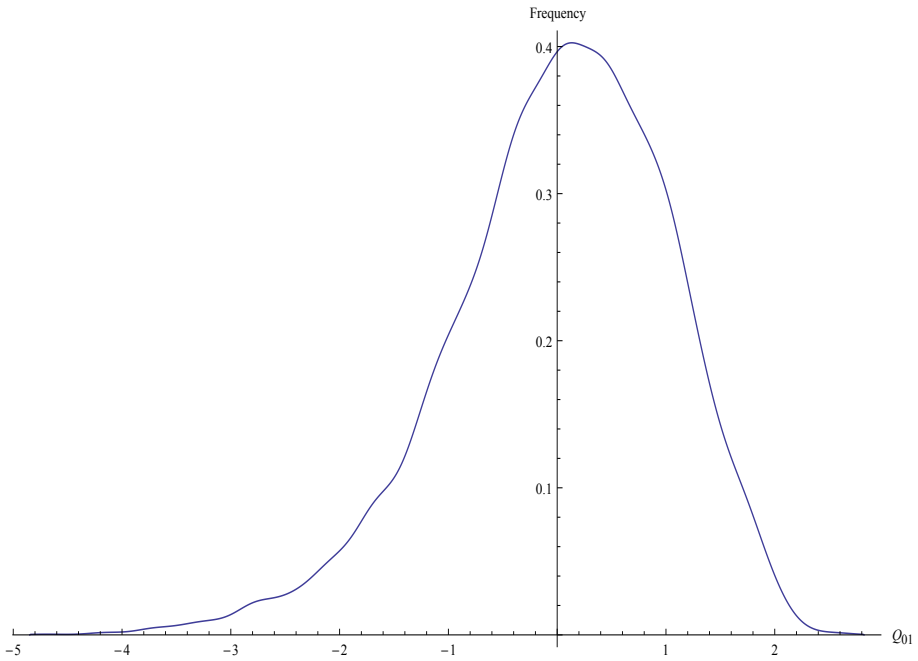

Figure 1. Distribution of Q_{01} values in the 10,000 samples generated.

These biases are represented through the appearance of extreme values in each sequence. Figure 2 shows the extreme values of the pre-calculated H_z^t distribution that cause such skewness to illustrate this event.

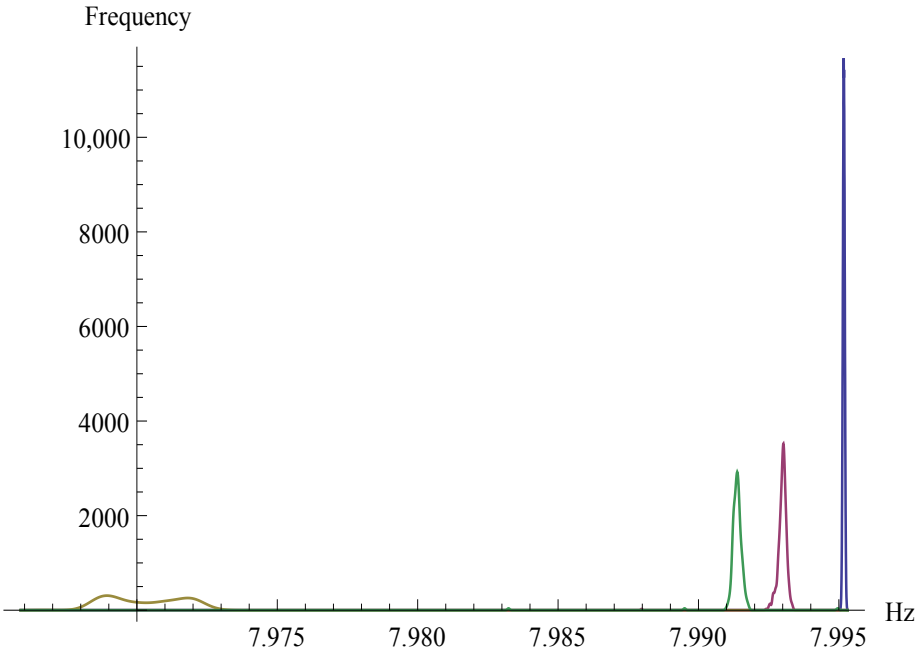

Figure 2. Representation of the extreme values of H_z^t in the pre-calculation stage.

As can be seen, three groups stand out to the left of extreme values. The first two groups of extreme values are caused by the first and second output bytes of RC4, which are highly biases and not evenly distributed [21,22]. Then a third group, which is caused by the existence of z_t bytes in the rest of the outputs, with $t > 1$, that has a high correlation with j. The last group is the remaining values of H_z^t.

Finally, for a significance level $\alpha = 0.01$, it was obtained that 233 of the 10,000 sequences of outputs analyzed do not satisfy that $Q_{01} > Z_\alpha = -2326$. In other words, the output sequences that provide more information about the j variable were detected. In this way, the Q-statistic is able to distinguish within a set of RC4 output sequences the most vulnerable to iterative probabilistic attacks.

6. Conclusions

A statistical criterion was proposed, which allows for distinguishing a set of sequences of outputs of RC4. This Q-statistic is based on the conditional entropies of j_t given the value z_t, known at each time t. It was experimentally verified that the proposed criterion could determine the existence of a class of output sequences more vulnerable to iterative probabilistic attacks. Future work intends to strengthen the proposed criterion by using the conditional probabilities $P(S_t|z_t)$, as well as to extend the criterion to the case in which the output of RC4 is not known, and only the ciphertext obtained with that output is known. Another result will be to investigate the possible adjustment of the distribution of the Q statistic to some of the known distributions and theoretically determine the lowest value of M for which it is effective.

Author Contributions: Conceptualization, E.J.M.-C. and C.M.L.-P.; methodology, E.J.M.-C., C.M.L.-P., O.R. and G.S.-G.; software, E.J.M.-C.; validation, E.J.M.-C., C.M.L.-P., O.R. and G.S.-G.; formal analysis, E.J.M.-C., C.M.L.-P., O.R. and G.S.-G.; investigation, E.J.M.-C., C.M.L.-P. and G.S.-G.; writing—original draft preparation, G.S.-G. and E.J.M.-C.; writing—review and editing, E.J.M.-C., C.M.L.-P., O.R. and G.S.-G.; visualization, E.J.M.-C.; supervision, E.J.M.-C., C.M.L.-P., O.R. and G.S.-G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data sharing not applicable.

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

References

1. Knudsen, L.R.; Meier, W.; Preneel, B.; Rijmen, V.; Verdoolaege, S. Analysis methods for (Alleged) RC4. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 1998; Volume 1514, pp. 327–341. [CrossRef]
2. Golić, J.D. Iterative probabilistic cryptanalysis of RC4 keystream generator. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2000; Volume 1841, pp. 220–233. [CrossRef]
3. Golic, J.D.; Morgari, G. Iterative Probabilistic Reconstruction of RC4 Internal States. *IACR Cryptol. ePrint Arch.* **2008**, *2008*, 348.
4. Paul, G.; Maitra, S. *RC4: Stream Cipher and Its Variants*; CRC Press: Boca Raton, FL, USA, 2011; pp. 1–281. [CrossRef]
5. RC4 Cipher Is No Longer Supported in Internet Explorer 11 or Microsoft Edge. Available online: https://support.microsoft.com/en-us/help/3151631/rc4-cipher-is-no-longer-supported-in-internet-explorer-11-or-microsoft (accessed on 18 July 2020)
6. SSL Configuration Required to Secure Oracle HTTP Server After Applying Security Patch Updates. Available online: https://support.oracle.com/knowledge/Middleware/2314658{_}1.html (accessed on 18 July 2020)
7. Satapathy, A.; Livingston, J. A Comprehensive Survey on SSL/ TLS and their Vulnerabilities. *Int. J. Comput. Appl.* **2016**, *153*, 31–38. [CrossRef]
8. Soundararajan, E.; Kumar, N.; Sivasankar, V.; Rajeswari, S. Performance analysis of security algorithms. In *Advances in Communication Systems and Networks*; Springer: Singapore, 2020; Volume 656, pp. 465–476. [CrossRef]
9. Jindal, P.; Makkar, S. Modified RC4 variants and their performance analysis. In *Microelectronics, Electromagnetics and tElecommunications*; Springer: Singapore, 2019; Volume 521, pp. 367–374. [CrossRef]
10. Parah, S.A.; Sheikh, J.A.; Akhoon, J.A.; Loan, N.A.; Bhat, G.M. Information hiding in edges: A high capacity information hiding technique using hybrid edge detection. *Multimed. Tools Appl.* **2018**, *77*, 185–207. [CrossRef]
11. Capó, E.J.M.; Cuellar, O.J.; Pérez, C.M.L.; Gómez, G.S. Evaluation of input—Output statistical dependence PRNGs by SAC. In Proceedings of the 2016 International Conference on Software Process Improvement (CIMPS), Aguascalientes, Mexico, 12–14 October 2016; pp. 1–6. [CrossRef]
12. Grosul, A.L.; Wallach, D.S. *A Related-Key Cryptanalysis of RC4*; Technical Report; Department of Computer Science, Rice University: Houston, TX, USA, 2000.
13. Matsui, M. Key collisions of the RC4 stream cipher. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5665, pp. 38–50. [CrossRef]
14. Chen, J.; Miyaji, A. How to find short RC4 colliding key pairs. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 7001, pp. 32–46. [CrossRef]
15. Tyagi, M.; Manoria, M.; Mishra, B. Effective data storage security with efficient computing in cloud. *Commun. Comput. Inf. Sci.* **2019**, *839*, 153–164. [CrossRef]
16. Dhiman, A.; Gupta, V.; Singh, D. Secure portable storage drive: Secure information storage. *Commun. Comput. Inf. Sci.* **2019**, *839*, 308–316. [CrossRef]
17. Nita, S.L.; Mihailescu, M.I.; Pau, V.C. Security and cryptographic challenges for authentication based on biometrics data. *Cryptography* **2018**, *2*, 39. [CrossRef]
18. Zelenoritskaya, A.V.; Ivanov, M.A.; Salikov, E.A. Possible Modifications of RC4 Stream Cipher. *Mech. Mach. Sci.* **2020**, *80*, 335–341. [CrossRef]
19. Jindal, P.; Singh, B. Optimization of the Security-Performance Tradeoff in RC4 Encryption Algorithm. *Wirel. Pers. Commun.* **2017**, *92*, 1221–1250. [CrossRef]
20. Cover, T.M. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 1999.
21. Pudovkina, M. *The Number of Initial States of the RC4 Cipher with the Same Cycle Structure*; Technical Report Mod L; Moscow Engineering Physics Institute (State University): Moscow, Russia, 2003.
22. Mironov, I. (Not so) random shuffles of RC4. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2442, pp. 304–319. [CrossRef]
23. Zhang, Z.; Zhang, X. A normal law for the plug-in estimator of entropy. *IEEE Trans. Inf. Theory* **2012**, *58*, 2745–2747. [CrossRef]
24. Miller, G. Note on the bias of information estimates. In *Information Theory in Psychology: Problems and Methods*; Free Press: Glencoe, IL, USA, 1955; pp: 95–100.
25. Basharin, G.P. On a statistical estimate for the entropy of a sequence of independent random variables. *Theory Probab. Appl.* **1959**, *4*, 333–336. [CrossRef]
26. Dodge, Y. *The Concise Encyclopedia of Statistics*; Springer Science & Business Media: Berlin Heidelberg, Germany, 2008. [CrossRef]
27. Van Den Broek, F.; Poll, E. A comparison of time-memory trade-off attacks on stream ciphers. In *International Conference on Cryptology in Africa*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 406–423.
28. Verdú, S. Empirical Estimation of Information Measures: A Literature Guide. *Entropy* **2019**, *21*, 720. [CrossRef] [PubMed]