Two-Stage Method for Information-Theoretically Secure Data Encryption

Wolfgang A. Halang[1], Li Ping[1], Maytiyanin Komkhao[2], and Sunantha Sodsee[3]

[1] Chair of Computer Engineering, Fernuniversität in Hagen, Germany
wolfgang.halang@fernuni-hagen.de
[2] Faculty of Science and Technology,
Rajamangala University of Technology Phra Nakhon, Bangkok, Thailand
maytiyanin.k@rmutp.ac.th
[3] Faculty of Information Technology,
King Mongkut's University of Technology North Bangkok, Thailand
sunanthas@kmutnb.ac.th

Abstract. It is known from information theory that eavesdropping and gaining unauthorised access to computers can be prevented by cryptography employing one-time keys. Regarding this fact, a practically feasible data encryption and user authentication scheme is presented, whose two stages are already information-theoretically secure each on its own. The first stage generates packet-long one-time keys by a chaos-theoretical method, and the second one adds redundancy in form of allowing to encrypt any plaintext by a randomly selected element out of a large set of possible ciphertexts. Obliterating the symbol boundaries in transmission units, this cryptosystem removes a toehold for cryptanalysis not addressed before.

Keywords: Unbreakable encryption, eavesdropping, secure communication, authentication, chaos theory.

1 Introduction

In information and communication technology, increasingly datasets of any size are exchanged between computers in form of streams via data networks. To guarantee the confidentiality of such messages' contents, a plentitude of methods to encrypt the data streams was developed [7]. Currently used encryption methods usually employ the same keys during longer periods of time, lending themselves to cryptanalytic attacks. It was shown, for instance, that the rather widespread asymmetrical RSA-cipher with keys 768 bits long has at least theoretically been broken. The symmetrical cryptosystem DES is already regarded as unsafe, too. Other ciphers such as 3DES or AES are still being considered safe, but only because the presently available computing power is insufficient to carry out simple brute-force attacks. In some countries law requires to deposit the keys used with certain agencies. Thus, these countries' secret services do not need any cryptanalysis whatsoever to spy out encrypted data.

H. Unger et al. (eds.), *Recent Advances in Information and Communication Technology 2015*,
Advances in Intelligent Systems and Computing 361, DOI: 10.1007/978-3-319-19024-2_16

In consequence, only perfectly secure one-time encryption appears to be feasible in the long run. Perfect security is achieved, if encryption of a plaintext yields with equal probability any possible ciphertext, and if it is absolutely impossible to conclude from the ciphertext to the plaintext in a systematic way. According to the theorem of Shannon [8] fundamental for information theory, a cryptosystem is regarded as perfectly safe only then, if the number of possible keys is at least as large as the number of possible messages. Hence, also the number of keys is at least as large as the one of possible ciphertexts which, in turn, must be at least as large as the number of possible plaintexts. Based on these considerations, in the sequel a novel system for data encryption and user authentication is presented, which works with one-time keys as long as packets to be stored or transmitted.

The method employed to generate one-time keys is a chaos-theoretical one. Mathematical chaos [9] is one of the most well-known and potentially useful classes of non-linear dynamics. The dynamical behaviour of non-linear systems has gained strong interest in recent decades. As a result, non-linearity has become a major topic in mathematics and engineering sciences. Although chaotic systems are governed by simple and low-order deterministic rules, their dynamics are random-like and complex. These characteristics let chaotic systems become potential candidates for sources of pseudo-randomness such as building blocks in cryptographical applications. Summaries of various corresponding research activities and designs can be found in [1,5]. Here, more complicated chaotic systems, viz. spatiotemporal ones, are utilised as sources of pseudo-randomness due to their good performance. In particular, coupled map lattices are adopted as such spatiotemporal systems, which are more complex than other chaotic systems and can serve as multiple sources of pseudo-randomness [6].

To yield perfectly safe encryption, one-time keys need to be truly random. Since deterministic methods as the one described in Section 2 are only able to produce sequences of pseudo-random bits, however, in Section 3 we propose to make up for this deficiency by adding a second encryption stage. This stage features ample redundancy by allowing freely selectable plaintext segments to be encrypted by elements randomly chosen out of large sets of possible ciphertexts, and it blurs the boundaries between data symbols encrypted together. Thus, it is made impossible to conclude from boundaries between data items in ciphertexts to the boundaries of data items in the resulting plaintexts, removing a toehold for cryptanalysis which has been neglected so far. This paper ends with stating the two-stage encryption and decryption algorithms and some considerations pointing to a wide range of options for implementation and for modifications during communication processes.

2 Spatiotemporal Chaos Yielding Pseudo-random Bits

In order to produce one-time keys for encryption purposes, a novel Multiple Pseudo-Random Bits Generator (MPRBG) based on spatiotemporal chaos was proposed and comprehensively studied [6]. By their very nature, spatiotemporal

chaotic systems are dynamical systems, which are often described by partial differential equations, coupled ordinary differential equations or Coupled Map Lattices (CMLs). These dynamical systems exhibit chaotic properties in both time and space. Among them, the ones described by CMLs are most widely used, due to their digital nature and favourable combination of computational complexity and representation of the original systems.

Spatiotemporal chaos is created in CMLs by local non-linear dynamics and spatial diffusion. By adopting various non-linear mappings for local chaos, and various discretised diffusion processes, which are also regarded as coupling, different forms of CMLs can be constructed. Commonly used are the logistic map as local map and nearest-neighbour coupling.

A general nearest-neighbour CML can be described as

$$x_{i+1,j} = (1-\epsilon)f(x_{i,j}) + \frac{\epsilon}{2}[f(x_{i,j+1}) + f(x_{i,j-1})], \tag{1}$$

where $i = 1, 2, ...$ is the time index, $j = 1, 2, ..., L$ with $L \geq 2$ is the lattice site index with a periodic boundary condition, f is a local chaotic map in the interval I, and $\epsilon \in (0, 1)$ is a coupling constant. Here, the logistic map is taken as local map, which is described by

$$f(x) = rx(1-x), \tag{2}$$

where $r \in (0, 4]$ is a constant. An example of the spatiotemporal chaos generated by Eqs. (1) and (2) is shown on the left side of Fig. 1.

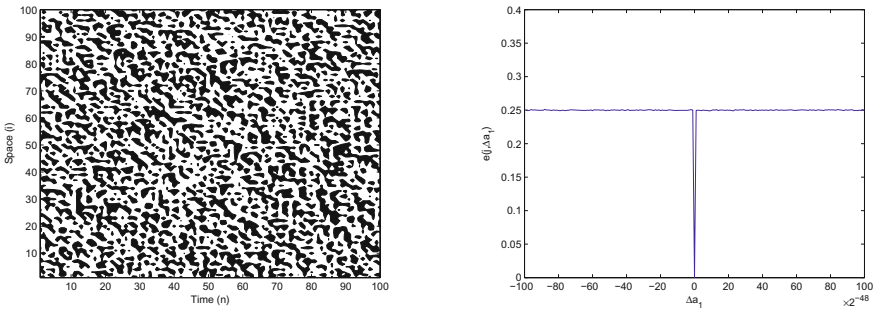

Fig. 1. Left: pattern of a CML with $\epsilon = 0.9$, $r = 4$ and $L = 100$; right: error function

Thus, a CML with L lattice sites can simultaneously generate L pseudo-random bit sequences by digitising the chaotic outputs of the lattice sites. The state variable $x_{i,j}$ of the j-th site can be regarded as a pseudo-random number, which means that $\{x_{i,j}\}_{i=1}^{\infty}$ is a Pseudo-Random Number sequence (PRNS), denoted by PRNS_j. Therefore, L PRNSs can simultaneously be generated from a CML of size L. Further, by digitising the PRNSs, i.e. by transforming the

sequence of real numbers to a binary sequence, Pseudo-Random Bit Sequences (PRBSs) can be obtained. Here, a PRBS is generated by concatenating the mantissae of the numbers $x_{i,j}$ in a certain floating-point representation [6].

In order to prevent the lattice sites from falling into synchrony, a criterion was derived by analysing the Lyapunov exponents $\lambda_j,\ j = 1, 2, \ldots, L$, of the CML, namely that λ_2 must be bigger than 0, i.e.

$$\lambda_1 + \ln[1 - \epsilon + \epsilon \cdot \cos(2\pi/L)] > 0, \tag{3}$$

or

$$r\left[1 - \epsilon\left(1 - \cos\frac{2\pi}{L}\right)\right] > 2. \tag{4}$$

When satisfying Condition (4), the CML with r, ϵ and L can exhibit chaotic behaviour without its sites falling into a state of synchronisation.

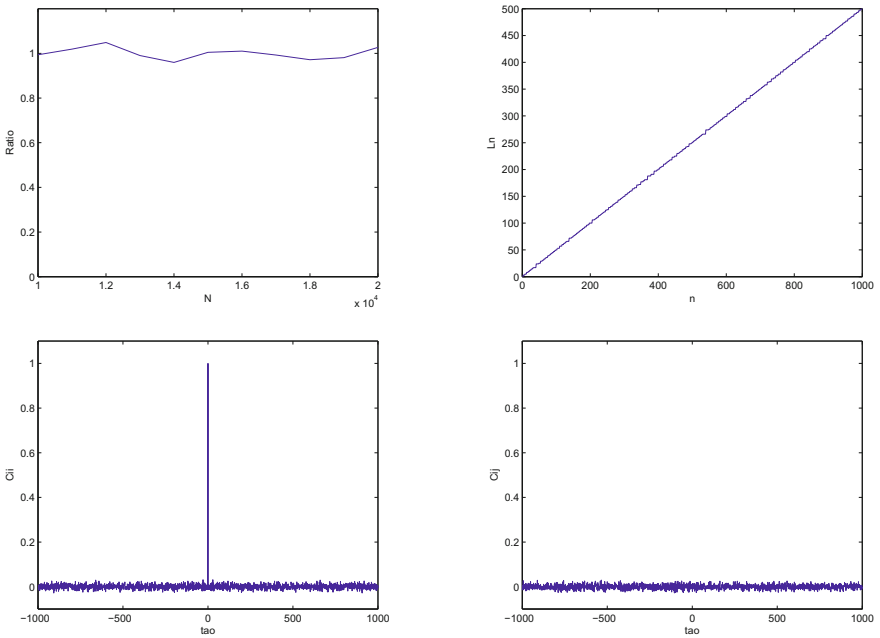

Fig. 2. Cryptographic properties of a pseudo-random bit sequence: 0:1 ratio, linear complexity, auto- and cross-correlation

The cryptographic properties of this approach were studied. The first one is a long period, which is ensured by spatiotemporal chaotic systems, although there exists a problem of short periods along with the chaotic orbits, when chaotic maps are realised in computers with finite precision. The period of CMLs with L lattices is about $10^{-0.4L} \cdot 2^{52\times 0.47L} \approx 10^{7L}$, and the period of the PRBSs generated by CML-MPRBGs is also about 10^{7L}. Therefore, when $L > 5$, the

period provided by CML-MPRBGs satisfies the basic cryptographic requirement, since a length of order $O(2^{100})$ is cryptographically long. The second property is balance, which means that a PRBS has a uniform distribution, i.e. with about the same numbers of 0's and 1's in the binary sequence. The third one is strong linear complexity. The forth property, that a PRBS has a cross-correlation close to zero, can be used to encrypt several plaintexts at one time. The last one is that a PRBS has a δ-like auto-correlation, which measures the extent of similarity between the PRBS and a shift of itself by some positions. These five properties have been investigated numerically by calculations as depicted in Fig. 2. Based on them, the novel scheme to generate multiple streams of pseudo-random bits promises to be advantageous for fast and secure encryption.

A one-way cyclically coupled logistic-map lattice with certain parameters has the best cryptographic properties among the six most simple pseudo-random bits generators [6]. Based on this, a stream cipher is designed, which carries out encryption as follows:

$$\begin{aligned} f(x_{i,j}, a_j) &= (3.9 + 0.1a_j)x_{i,j}(1 - x_{i,j}), \\ x_{i+1,j} &= (1 - \epsilon)f(x_{i,j}, a_j) + \epsilon f(x_{i,j-1}, a_{j-1}), \\ K_{i,j} &= \mathrm{int}[x_{i,j} \times 2^u] \bmod 2^v, \\ C_{i,j} &= M_{i,j} \oplus K_{i,j}, \qquad j = 1, ..., L,\ i \in \mathbb{N} \end{aligned} \tag{5}$$

where the index 0 is to be replaced by L for the cyclic coupling of lattice sites, $u, v \in \mathbb{N}$ suitably chosen, $K_{i,j}$, $M_{i,j}$ and $C_{i,j}$ are keystream, plaintext and ciphertext, respectively, and $\oplus$ denotes bitwise antivalence. Actually, the CML serves as a PRBG to produce L keystreams by applying the algebraic operations *int* and *mod* on the outputs of the CML. Plaintexts are subjected to bitwise exclusive or with keystreams to produce ciphertext. Encryption parameters are assumed as $a_j \in [0, 1]$, denoted in vector form as $\mathbf{a} = \{a_1, a_2, ..., a_L\}$.

The configuration and parameters of decryption are the same as those of encryption, which is described as

$$\begin{aligned} f(y_{i,j}, a'_j) &= (3.9 + 0.1a'_j)y_{i,j}(1 - y_{i,j}), \\ y_{i+1,j} &= (1 - \epsilon)f(y_{i,j}, a'_j) + \epsilon f(y_{i,j-1}, a'_{j-1}), \\ K'_{i,j} &= \mathrm{int}[y_{i,j} \times 2^u] \bmod 2^v, \\ M'_{i,j} &= C_{i,j} \oplus K'_{i,j}, \qquad j = 1, ..., L,\ i \in \mathbb{N} \end{aligned} \tag{6}$$

where $a'_j \in [0, 1]$ are decryption parameters, denoted as $\mathbf{a}' = \{a'_1, a'_2, ..., a'_L\}$. When $\mathbf{a}' = \mathbf{a}$ and $y_{0,j} = x_{0,j}$ holds for the seed values, these two CMLs are synchronised, i.e. $y_{i,j} = x_{i,j}$, $i \in \mathbb{N}$, thus producing identical keystreams, $K'_{i,j} = K_{i,j}$. As a result, plaintext is decrypted, $M'_{i,j} = M_{i,j}$.

For the keystreams in this cipher to have proper statistical properties, the parameters a_j, $j = 1, 2, ..., L$, are set to guarantee that the logistic map's coefficient r in Eq. (2) falls into the range $[3.9, 4.0]$, and ϵ is fixed as 0.95. The quantity v is assumed as 32 for the following reasons, and u is selected accordingly, e.g. as 52 when double-precision floating-point arithmetic is used. First, the leading 4 bits are discarded for their bad statistical properties. Then, the smaller v is,

the harder it is to break the cipher with known-plaintext attacks. Finally, from the implementation point of view, the larger v is, the more efficient the cipher will be. Therefore, a trade-off between efficiency and security leads to fix v as 32 by considering that common computers work with numbers 32 bits or 64 bits wide. When determining L, the following considerations are important. There is no evident influence of L on the cryptographic properties of the keystream except for its period equal to about 10^{7L}, there is no influence on the encryption speed either, and the cost of breaking the cipher is about 2^{40L}. Therefore, in investigating a concrete cipher thereafter, L is assumed as 4 in order to let the keystream's period be 10^{28} and the cost of breaking the cipher be up to 2^{160}, which are suitable choices from the cryptographic point of view.

A keyspace is defined as a set of all possible keys, which should be studied in depth when designing a cipher. An error function is used here to determine the size of the cipher's keyspace. When $\mathbf{a}' \neq \mathbf{a}$, the decrypted plaintext, $M'_{i,j}$, can deviate from the original one, $M_{i,j}$. The error function is defined as

$$\begin{aligned} &e(j, \Delta\mathbf{a}_t) = \tfrac{1}{T}\textstyle\sum_{i=1}^{T} |m'_{i,j} - m_{i,j}|, j = 1, 2, ..., L, \\ &m'_{i,j} = 2^{-32} \cdot M'_{i,j}, \quad m_{i,j} = 2^{-32} \cdot M_{i,j}, \end{aligned} \tag{7}$$

where $\Delta\mathbf{a}_t = \{\Delta a_1, \Delta a_2, ..., \Delta a_t\}$ $(\Delta a_j = a'_j - a_j,\ j = 1, 2, ..., t,\ t \leq L)$, and T is the length of encryption. The error function vs. Δa_1 with $T = 10^5$ is plotted on the right side of Fig. 1. It is shown that the error function is not equal to zero but 0.25, even if Δa_1 takes on an extremely small value 2^{-47}. In other words, the parameter a'_1 is sensitive to any differences equal to or larger than 2^{-47}. Similarly, the error functions of Δa_j $(j = 2, 3, ..., L)$ were computed, indicating that the parameters a'_j $(j = 2, 3, ..., L)$ are also sensitive to any differences equal to or larger than 2^{-47}. Therefore, the keyspace is 2^{47L}.

Since ciphertext is generated by direct bitwise application of the antivalence operator between plaintext and keystream, the cryptographic properties of the keystream have significant effects on the security of the cipher. Owing to the symmetric configuration of the CML, all keystreams have similar cryptographic properties. Some cryptographic properties of a keystream among the L ones, such as probability distribution, auto-correlation and run probability, were investigated numerically. In summary, L keystreams have satisfactory random-like statistic properties. Moreover, the security of the cipher was evaluated by investigating its confusion and diffusion properties and using various typical attacks, such as the error function attack, the differential attack, the known-plaintext attack, the brute-force attack and the chosen-plaintext/ciphertext attack.

3 Most General Form of Encrypting Bit Patterns

All known cryptographic methods subject the data elements to be transmitted, may that be bits, alphanumerical characters or bytes containing binary data, may they be single or in groups, always as unchanged entities to encryption. Shannon's [8] information-theoretical model of cryptosystems is founded

on this restrictive basic assumption as well. Consequently, information such as the boundaries between data elements and their number perpetuates observably and not encrypted into the ciphertext: as a rule, to any plaintext symbol there corresponds exactly one ciphertext symbol. Since even block ciphers seldom work with data entities exceeding 256 bits, the symbols in plaintexts and in ciphertexts are ordered in the same sequences or, at least, their positions lie very close together. Thus, corresponding symbols in plaintext and ciphertext can rather easily be associated with one another.

As this feature facilitates code-breaking, a counter-acting enhancement for cryptographic systems was devised [3]. The method's fundamental idea is based on the observation that, ultimately, technical realisations represent all symbols in binary encodings. Correspondingly, for encryption the most general among all possible forms of replacing one bit pattern by another one is employed. This allows to blur the boundaries between the plaintext symbols, and to use several, randomly selected encryptions for a single bit pattern, having more bit positions in the ciphertext than in the plaintext.

In a state t of a communication process, the number m_t of bit positions to be encrypted according to [3] is determined by an arbitrarily selectable method. It is decisive that the parameter m_t is different from the number of bit positions k encoding the plaintext alphabet. Thereby the boundaries between the plaintext symbols are annihilated. Then, for m_t bits each in a stream, an encryption with n bit positions is determined by means of a state-dependent *relation*

$$R_t \subset \{0,1\}^{m_t} \times \{0,1\}^n. \tag{8}$$

Here, the parameter n may not be smaller than m_t, as information would get lost otherwise, and it should not be equal to m_t either, in order to prevent the disadvantages mentioned above. Choosing $m_t \neq k$ and $n > m_t$ inherently ensures that it is not easily possible anymore to conclude from the boundaries between the ciphertext symbols on the ones between the plaintext symbols.

Contrary to the conventional cryptographic methods, the relation R_t does not need to be a mapping: it is even desirable that with any element in $\{0,1\}^{m_t}$ as many elements of $\{0,1\}^n$ as possible are related by R_t, allowing to randomly select among them one as encryption. For $n > m_t$, the set of possible encryption elements is embedded in a considerably larger image set, significantly impeding code analysis for an attacker. Moreover, every element in $\{0,1\}^n$ should be a valid cipher of an element in $\{0,1\}^{m_t}$, to completely exhaust the encryption possibilities available. Then, unique decipherability is given, if and only if the inverse relation is a surjective (onto) mapping:

$$R_t^{-1} : \{0,1\}^n \longrightarrow \{0,1\}^{m_t}. \tag{9}$$

Different from Kerckhoffs' [4] principle, this decryption function is not known publicly – and the relation R_t used for encryption is not only publicly unknown, but no function either. Publicly known is only, that R_t^{-1} is a totally arbitrary mapping among all possible ones mapping the finite set $\{0,1\}^n$ onto another finite set $\{0,1\}^{m_t}$.

The number of all possible relations $R_t \subset \{0,1\}^{m_t} \times \{0,1\}^n$, for which R_t^{-1} is a surjective mapping, amounts to $\frac{2^n!}{(2^n-2^{m_t})!}$. The set of these relations comprises, among others, all possibilities to permutate bits in their respective positions, to insert $n-m_t$ redundant bits, each of which may have either one of both possible values 0 or 1, at $\binom{2^n}{2^{n-m_t}}$ positions in the output bit patterns as well as to link the bit positions of the encryption elements by the most general computations.

4 Implementation of Two-Stage Data Encryption

A system for encrypting data packets to be transmitted over wired or wireless communication networks is to be devised now, which should be both information-theoretically secure and practically feasible. Since the PRBG described in Section 2 does not produce truly random bit sequences, we combine it with the method of Section 3 in defining the following two-stage encryption algorithm.

1. As many iterations of the PRBG in Eq. (5), i.e. recurrent floating-point calculations, are carried out as required for the concatenated resulting bit sequences to match or exceed the length of a data packet to be encrypted. Then, bitwise antivalence is formed between the packet and the bit sequence serving as one-time key.
2. The bit string resulting from step (1) and having the same length as the original packet is further processed by the method of Section 3, i.e. the bit string is partitioned into segments of length m_t (cp. Fig. 3), and for each segment an image is randomly selected among the images the segment relates to by relation R_t. The concatenation of all images thus determined, and each being n bits long, constitutes the packet's final enciphering ready for transmission.

To decrypt such a data packet, a receiver carries out the inverse operations expected to be applicable to the packet in the current state of a communication.

1. A packet received is partitioned into n bits long segments, each of which is subjected to the mapping R_t^{-1}. The results are concatenated.
2. As many iterations of the PRBG in Eq. (6) are carried out as required for the concatenated resulting bit sequences to match or exceed the length of the data packet to be decrypted. Bitwise antivalence is finally formed between the packet and the bit sequence.

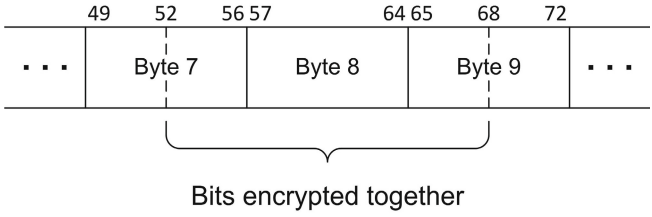

Fig. 3. Extracting bit strings from a data packet

The condition for the above algorithms to work correctly is that a sender's and a receiver's PRBG run in synchrony, i.e. their parameters have the same values and their iteration counts are equal. Since a communicating device usually acts in turn as sender and receiver, it needs to maintain both the relation R_t and the inverse R_t^{-1} for each of its communication partners.

Breaking this cipher were only possible, if an eavesdropper had such an amount of ciphertexts available as required by pertaining analyses – totally disregarding the necessary computational power. Among the following – non-exhaustive – variety of implementation options there are some further measures preventing sufficiently long encipherings, generated with certain choices of parameter sets and encryption relations, to arise in the first place.

- The order of the two steps applying one-time keys and the encryption relation may be reversed in the algorithms above.
- The seed and parameter values of the PRBG running in sender and receiver may be modified frequently and irregularly. To implement this, one of the communication partners may serve as master and may employ a physical phenomenon with truly random behaviour, e.g. measuring white noise. At randomly determined points in time, either new seed and parameter values or just array indices may be included into the data packets transmitted. The latter case resembles the iTAN procedure of on-line banking. The indices would identify locations in read-only memory modules, whose production, transport and installation would typically represent a confidential and authentic channel to transfer secret information.
- The selection of images among the ones provided by relation R_t may be based on a physical phenomenon with truly random behaviour.
- Communicating units may, during operation at randomly selected points in time, sufficiently often vary the parameter m_t between 1 and an installation-dependent upper bound, thus modifying the relation R_t correspondingly.
- Similarly, communicating units may turn to use a completely different encryption relation R_t provided in real-only memory, which only requires to transmit its identifier and some parameter values.
- A simplified version of the relation-based encryption may entail the contents of R_t and other parameters to be supplied in form of pseudo-random bit sequences as well. A pertaining protocol may co-ordinate that both sender and receiver proceed, rather frequently at random instants, from one state to the next. In the course of a state transition, the relation R_t and the parameter m_t are re-defined. For co-ordination, as few details as possible should be transmitted between the communicating units for reasons of confidentiality.

Owing to its high complexity, the encryption procedure lends itself for authentication purposes in a straightforward way. To authenticate a packet, the receiver just needs to check for expected values in certain data fields. The bit patterns found there will be different if the packet does not come from the correct source, or anything has gone wrong.

5 Conclusion

Currently applied cryptosystems to secure confidential data transmission have either already been broken or are expected to be broken soon. Moreover, in certain countries their keys need to be escrowed with government agencies. In order to prevent eavesdropping and gaining unauthorised access to computers, the method of choice is, therefore, information-theoretically secure one-time encryption. Since providing a very large number of rather long one-time keys is practically infeasible, an efficient chaos-theoretical approach for the continuous generation of pseudo-random bit strings to be employed as one-time keys was presented. To compensate for this method's deficiency, viz. the lack of genuine randomness, it was combined with encrypting a bit pattern by a longer one selected truly at random within a larger set of possible ciphers.

To encrypt not by means of a bijective, i.e. invertible, function, but by a relation with a surjective mapping as inverse, is already known from the patent [2]. There the symbols of a plaintext alphabet are bijectively mapped onto equivalence classes of symbols in an image set of generally higher cardinality. To encrypt a plaintext symbol, out of the equivalence class corresponding to it in the image set an image symbol is selected, and that either randomly or so that the enciphering becomes as invulnerable by statistical methods as possible. Whereas according to [2] plaintext symbols are replaced one-to-one by image symbols in ciphertexts, the second stage of the method presented here blurs the boundaries between data items encrypted together, rendering it impossible to conclude from boundaries between data items in ciphertexts to the boundaries of data items in plaintexts. Thus, a toehold for cryptanalysis left open by a silent assumption in Shannon's communication theory was eliminated.

References

1. Álvarez, G., Li, S.J.: Some Basic Cryptographic Requirements for Chaos-based Cryptosystems. International Journal of Bifurcation and Chaos 16(8), 2129–2151 (2006)
2. Günther, C.-G.: Ein universelles homophones Codierverfahren, German patent 39 04 831 (1989)
3. Halang, W.A., Komkhao, M., Sodsee, S.: A Stream Cipher Obliterating Data Element Boundaries. Thai patent registration 140-100-1271 (2014)
4. Kerckhoffs, A.: La cryptographie militaire. Journal des Sciences Militaires 9. Serie (1883)
5. Kocarev, L.: Chaos-based Cryptography: A Brief Overview. IEEE Circuits and Systems Magazine 1, 6–21 (2001)
6. Li, P.: Spatiotemporal Chaos-based Multimedia Cryptosystems. Fortschr.-Ber. VDI Reihe 10 Nr. 777. VDI-Verlag, Düsseldorf (2007)
7. Menezes, A.J., van Oorschot, P.C., Vanstone, S.A.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1997)
8. Shannon, C.E.: Communication Theory of Secrecy Systems. Bell System Technical Journal 28, 656–715 (1949)
9. Silva, C.P.: A Survey of Chaos and its Applications. In: IEEE MTT-S International Microwave Symposium Digest, vol. 3, pp. 1871–1874 (1996)