Novel transcendental encryption algorithm using BBP formula

Alex-Pauline Poudade
Lycée Louis-le-Grand, Paris, France
November 13th, 2022, rev. 1.0

Abstract: This paper discusses a new way to encrypt data using transcendental properties discovered through mathematical Bailey–Borwein–Plouffe formula (BBP). In regard to pseudo-stochastic computer methods, it enables a stronger non-linear model close to True number generators (TRNG) resistance without need for physical prior transmissions of initial stochastic patterns. Soleau envelope European deposit number: DSO2017001085 - deposit reference 26081971181200533 2017 National Institute of Industrial Property (INPI) February 2, 2017

Keywords: TRNG, stochastic process, pseudo-randomness, Pi, BBP, Bailey–Borwein–Plouffe formula, DSO, Soleau, INPI

JEL Classifications Numbers: C6, C63, C65, C69, F59, K24.

OEIS: (provisional only) N/a

REFERENCES: orcid.org/0000-0003-3037-1091

1 Introduction

This paper focuses on explaining the Transcrypt transcendental encryption system.

The absolute encryption algorithm consists in communicating a truly random physical band of randomness to its correspondent beforehand and encrypting the original message with this stochastic pattern by adding or superimposing it on the message, in the possession of both parties. These solutions exist but remain out of reach of the general public because they are expensive and require rigorous and heavy logistics of physical transport at the same time as a physical device of generation of randomness.

This algorithm is optimal because the algorithm is chance. It cannot be recreated, by definition.

The second-best solution consists in generating this true randomness (or *True number generator* TRNG in opposition to the pseudo-randomness) remotely on a computer, a means allowing a priori only the pseudo-random. Indeed, the number Pi, because of its transcendental property, allows to generate, without any additional cost linked to the physical transport or to the physical generation, a sequence of numbers without any repetition (thus with exactly the same level of entropy as the TRNG) and to infinity.

Above all, this algorithm is synonymous with a consumer solution.

Until September 19, 1995, it was impossible to implement this algorithm because to calculate the *n-th* digit after the decimal point of the number Pi, it was necessary to have previously calculated the n-1 digit. Only supercomputers could do it and the computation times were unusable.

But this has become possible since the BBP formula (or Bailey-Borwein-Plouffe formula) which allows to calculate the *n-th* digit after the decimal point of the number Pi without having to calculate the previous ones and using very little memory and time.

2 Presentation

Transcrypt (contraction of *Transcendental Crypting*) is a new and innovative point-to-point encryption algorithm, based on a discovery about a property of the transcendental number Pi, in 1995, by the French-Canadian mathematician Simon Plouffe.

Among the five criteria (confidentiality, integrity, availability, non-repudiation and authentication) of the security of an information system, Transcrypt only addresses confidentiality.

The Transcrypt transcendental encryption principle is the subject of a Soleau envelope deposit by Miss Alex-Pauline Poudade (national deposit number: DSO2017001085 and deposit reference 26081971181200533 2017) at the National Institute of Industrial Property (INPI) since February 2, 2017. Its explicit and written authorization is required for any use for profit.

Transcrypt is openly licensed under the Creative Commons CC-BY-NC-SA free distribution license.

3 Encryption protocol

Scenario/protocol: Person A wishes to send a clear message M_{1p} in encrypted form to person B who replies.

1- A gets the delta $\Delta_{1c}$of the displacement corresponding to the position of the beginning of the strip in the decimals of Pi, saved.

2- A new delta calculation Δ_{2c} for $M_{2c}$of B and applies one or exclusive XOR $\Delta_{2c}= b_0 * b_2 \otimes b_1 * b_4$of the first bytes of $M_{1p}$and saves it

3- A transforms the message $M_{1p}$from binary to hexadecimal

4- A gets the length L_{1p} in bytes of the message M_{1p}

5- A obtains by BBP a strip of π of length L_{1p} starting position Δ_{1c}

6- A applies one or exclusive XOR $M_{1c} = M_{1p} \otimes \pi[\Delta_{1c}, \Delta_{1c} + L_{1p}]$

7- A communicates M_{1c} to B

8- B gets the delta $\Delta_{1c}$of the displacement corresponding to the position of the beginning of the strip in the decimals of Pi, saved.

9- B gets the length $L_{1c} = L_{1p}$ in bytes of the message M_{1c}

10- B obtains by BBP a strip of π of length L_{1c} starting position Δ_{1c}

11- B applies one or exclusive XOR $M_{1p} = M_{1c} \otimes \pi[\Delta_{1c}, \Delta_{1c} + L_{1c}]$

12- B transforms the message $M_{1p}$from hexadecimal base to binary base

13- B calculates a new delta Δ_{2c} and applies one or exclusive XOR $\Delta_{3c}= b_0 * b_2 \otimes b_1 * b_4$of the first bytes of $M_{2p}$and saves it

14- B gets the delta $\Delta_{1c}$of the displacement corresponding to the position of the beginning of the strip in the decimals of Pi, saved.

15- B calculates a new delta Δ_{2c} for $M_{3c}$of A and applies one or exclusive XOR $\Delta_{2c}= b_0 * b_2 \otimes b_1 * b_4$of the first bytes of $M_{2p}$and saves it

16- B transforms the message $M_{2p}$from binary to hexadecimal

17- B gets the length L_{2p} in bytes of the message M_{2p}

18- B obtains by BBP a strip of π of length L_{2p} starting position Δ_{2c}

19- B applies one or exclusive XOR $M_{2c} = M_{2p} \otimes \pi[\Delta_{2c}, \Delta_{2c} + L_{2p}]$

20- B communicates M_{2c} to A

21-A gets the delta $\Delta_{2c}$of the displacement corresponding to the position of the beginning of the strip in the decimals of Pi, saved.

22-A gets the length $L_{2c} = L_{2p}$ in bytes of the message M_{2c}

23-A obtains by BBP a strip of π of length L_{2c} starting position Δ_{2c}

24-A applies one or exclusive XOR $M_{2p} = M_{2c} \otimes \pi[\Delta_{2c}, \Delta_{2c} + L_{2c}]$

25-A transforms the message $M_{2p}$from hexadecimal to binary

1. A calculates a new delta Δ_{3c} and applies one or exclusive XOR $\Delta_{3c}= b_0 * b_2 \otimes b_1 * b_4$to the first bytes of $M_{2p}$and saves it

NOTE: a Proof of Concept (*POC)* in Python version 3 and ECMAScript is provided.

References

Wikipedia, Bailey–Borwein–Plouffe formula
https://en.wikipedia.org/wiki/Bailey%E2%80%93Borwein%E2%80%93Plouffe_formula

Pi https://en.wikipedia.org/wiki/Pi

Transcendental number https://en.wikipedia.org/wiki/Transcendental_number

Simon Plouffe https://en.wikipedia.org/wiki/Simon_Plouffe discoverer of BBP formula

Lincoln Stein "W3C The World Wide Web Security FAQ and Client Side Security"
https://www.w3.org/Security/Faq/wwwsf2.html June 1998 & on

"Crypto Talk at 27C 3: Is the SSLiverse a safe place? Day 2, 16:00, Saal 2"
https://events.ccc.de/2010/12/28/is-the-ssliverse-a-safe-place/ December 28, 2010

MD5.
https:
<BibTeX>
<RIS>
<EndNote XML>