




Problems and solutions from the fourth International Students' Olympiad in Cryptography (NSUCRYPTO)

A. Gorodilova , S. Agievich, C. Carlet, E. Gorkunov, V. Idrisova, N. Kolomeec, A. Kutsenko, S. Nikova, A. Oblaukhov, S. Picek, B. Preneel, V. Rijmen, and N. Tokareva

ABSTRACT

Mathematical problems and their solutions from the fourth International Students' Olympiad in cryptography (NSUCRYPTO-2017) are presented. We consider problems related to attacks on ciphers and hash functions, cryptographic Boolean functions, linear branch numbers, addition chains, and error correction codes, among others. We discuss several open problems involving the algebraic structures of cryptographic functions, useful proof-of-work algorithms, the Boolean hidden shift problem, and quantum computing.

KEYWORDS

Addition chains; error correction codes; hash functions; linear branch numbers; masking; NSUCRYPTO; Olympiad; proof-of-work

Introduction

The International Students' Olympiad in Cryptography (NSUCRYPTO) was held for the fourth time in 2017. The idea of the Olympiad was born at Novosibirsk State University, which is located in the world-famous scientific heart of Siberia—Akademgorodok. Now, the Olympiad Program Committee includes specialists from Belgium, France, the Netherlands, the United States of America, Norway, India, Belarus, and Russia. At the same time, the geography of participants is expanding year by year: there were more than 1,300 participants from 38 countries during 2014–2017.

Let us shortly formulate the format of the Olympiad (all information can be found on the official website, nsucrypto.nsu.ru). One of the Olympiad's ideas is that everyone can participate: school students, university students, and even professionals! Each participant chooses his/her category when registering on the Olympiad website. The Olympiad consists of two independent internet rounds; the first one is individual while the second round is team-oriented. The first round is divided into two sections, “A” for school students and “B” for university students and professionals. Participants read the Olympiad problems and send their solutions using the Olympiad website. The language of the Olympiad is English.

Every year participants are offered the chance to solve several problems of differing complexity at the intersection of mathematics and cryptography. Another feature of the Olympiad is that it not only includes interesting tasks with known solutions but also offers unsolved problems in this area. All the open problems stated during the Olympiad's history can be found at nsucrypto.nsu.ru/unsolved-problems. On the website we also mark the current status of each problem. For example, the problem "Algebraic immunity" (2016) was completely solved during the Olympiad and a partial solution for the problem "A secret sharing" (2014) was proposed in Geut et al. 2017. We invite everybody who has ideas on how to solve the problems to send your solutions to us!

What was surprising and very pleasant for us this year was that the NSUCRYPTO Olympiad can even change people's professional lives! The following are some examples of the feedback we have received over the years:

"Without having a 3rd ranking to go for PhD, I would give up in cryptography due to the complex mathematics involve."—Duc Tri Nguyen (Vietnam, third place in 2016 and 2017)

"When we joined the competition, we just wanted to know where we were with our knowledge, wanted to test ourselves with our self-study Crypto, and wanted to improve our math in Cryptography."—Quan Doan (Vietnam, third place in 2016 and 2017)

"The problems contain much knowledge not only from mathematics and cryptography, but also from many other fields such as art."—Ren Zhang Liu (China, first place in 2015)

"I sometimes find myself reading too much and this competition is a great way of putting knowledge into practice by solving fun tasks."—Dragos Alin Rotaru (United Kingdom, third place in 2016)

"It's a contest that leaves you wanting to spend more time on it after the deadline, just to work out the questions you didn't get."—Robert Spencer (United Kingdom, first place in 2016, second place in 2017)

"Participation in the Olympiad offers you an excellent opportunity to try yourself as a codebreaker and a cryptographer."—Anna Taranenko (Russia, first place in 2014, second place in 2015 and 2016, and honorable diploma in 2017)

"Sometimes you spend hours trying to solve a problem, sometimes it seems that it's impossible to solve it. But when you find a solution, usually so obvious, you experience an incomparable sense of delight. My hat's off to the person who thinks up such interesting tasks!"—Evgeniya Ishchukova (Russia, third place in 2016, honorable diplomas in 2015 and 2017).

A complete list of comments can be found at nsucrypto.nsu.ru/feedbacks.

Table 1. Problems from the first round.

A: School Section		
N	Problem Title	Maximum Score
1	PIN code	4
2	Chests with treasure	4
3	A numerical rebus	4
4	Timing attack	4
5	The shortest addition chain	4
6	A music lover	4
B: Student Section		
N	Problem Title	Maximum Score
1	Timing attack	4
2	Treasure chests	4
3	A music lover	4
4	An infinite set of collisions	4
5	One more parameter	10
6	Scientists	8
7	Masking	10

The paper is organized as follows. We start with the structure of problems structure at the Olympiad. Then we present formulations of all the problems stated during the Olympiad and give their detailed solutions. Finally, we publish the list of NSUCRYPTO-2017 winners.

Mathematical problems from the previous International Olympiads (NSUCRYPTO-2014, NSUCRYPTO-2015, and NSUCRYPTO-2016) can be found in Agievich et al. 2015, 2017 and Tokareva et al. 2018, respectively.

Problem structure of the Olympiad

There were 16 problems stated during the Olympiad, and some of them were included in both rounds (Tables 1 and 2). The school section of the first round consisted of six problems, whereas the student section contained seven problems. Two problems were common to both sections. The second round was composed of eleven problems; they were common for all the participants. Three problems from the second round were marked as “unsolved” (if someone proposed a complete solution to this problem, he/she would be awarded special prizes from the Program Committee).

Problems and their solutions

In this section we formulate all the problems from NSUCRYPTO-2017 and present their detailed solutions while paying attention to solutions proposed by the participants.

Problem: “PIN code”

Formulation

A PIN code $P = \overline{p_1 p_2 \dots p_n}$ is an arbitrary number consisting of a finite number of pairwise different decimal digits in ascending order

Table 2. Problems from the second round.

N	Problem Title	Maximum Score
1	The image set	Unsolved
2	TwinPeaks	8
3	An addition chain	8
4	Hash function FNV2	8
5	A music lover	4
6	Boolean hidden shift and quantum computings	Unsolved
7	One more parameter	10
8	Scientists	8
9	Masking	10
10	PIN code	4
11	Useful proof-of-work for blockchains	Unsolved

$(p_1 < p_2 < \dots < p_n)$. Bob got his personal PIN code from the bank, but he decided that the code is not secure enough and changed it in the following way:

1. Bob multiplied his PIN code P by 999 and obtained the number $A = \overline{a_1 a_2 \dots a_m}$;
2. Then he found the sum of all digits of A : $a_1 + a_2 + \dots + a_m = S = \overline{s_1 s_2 \dots s_k}$;
3. Finally, he took all digits (starting from zero) smaller than s_1 , sorted them in ascending order, and then inserted them between digits s_1 and s_2 in the number S . The resulting number P' is Bob's new PIN code.

For example, if S was 345 then after insertion we obtain $P' = 301245$.

Find the new code P' !

Remarks. By $\overline{p_1 p_2 \dots p_n}$ we mean that p_1, p_2, \dots, p_n are decimal digits and all digits over the bar form a decimal number.

Solution

Let $P = \overline{p_1 p_2 \dots p_n}$ for some positive integer n . Let us note that P multiplied by 999 is the same thing as P multiplied by 1,000 minus P ; that is, shifting the number P three positions to the left minus itself. Let us consider this subtraction:

$$\begin{array}{r}
 p_1 \quad p_2 \quad p_3 \quad p_4 \quad \cdots \quad p_n \quad 0 \quad 0 \quad 0 \\
 - \quad \quad \quad p_1 \quad \cdots \quad p_{n-3} \quad p_{n-2} \quad p_{n-1} \quad p_n \\
 \hline
 \end{array}$$

Since $p_1 < p_2 < \dots < p_n$ by definition, we have $p_{n-3} < p_n$. Therefore, we borrow a unit only from p_n among p_1, \dots, p_n . Thus, the sum of digits of this difference is equal to

$$\begin{aligned}
 & (10 - p_n) + (9 - p_{n-1}) + (9 - p_{n-2}) + (p_{n-1} - p_{n-3}) \\
 & + (p_{n-1} - p_{n-4}) + \dots + (p_4 - p_1) + p_3 + p_2 + p_1 = 27.
 \end{aligned}$$

So, Bob's new PIN code is 2017.

Many great and compact solutions were sent by the participants. The best solution in the school section was by Lenart Bucar (Gimnazija Bežigrad, Grosuplje, Slovenia). Also, we want to note a detailed solution sent by Ivan Baksheev (Gymnasium 6, Novosibirsk, Russia).

Problem: “Chests with treasure”

Formulation

We have three closed chests. Some of them contain treasure (diamonds, gold coins, and bitcoins), but we do not know which ones. A parrot knows which chests contain treasure and which do not; he agrees to answer questions with “yes” or “no.” He may possibly lie in his answers, but not more than once. List six questions such that it is possible to deduce from the parrot’s answers which chests contain treasure and which do not.

Solution

Let us mark the chest with zero if it does not contain the treasure and with one if it does. Similarly, let us write down the parrot’s answers to our questions as ones for “yes” and zeros for “no.” Now the state of the chests is encoded with three bits of information, and the parrot’s answers give us six bits of information. Taking into account that the parrot may or may not lie in one of his answers, one or none of the bits among these six can be faulty.

So, our goal is to devise six questions in such a way that six bits of obtained information can be uniquely decoded to exactly one three-bit state of chests, even if one bit of answers is not right. In terms of coding theory, we need to construct an error-correcting code that corrects one error. In simple terms, we need to map every one of $2^3 = 8$ chest states to a Boolean vector of length six in such a way that even if one bit in any of these image vectors gets flipped, we can uniquely determine which vector it was before the error occurred.

To find these eight vectors it is sufficient to find eight vectors of length six such that every two of them are different in at least three bits. Even if we receive one of these vectors with a flipped bit, it is still possible to determine which of these eight vectors it was before the error. These eight vectors can be found manually. One example is shown below:

(000) \rightarrow (000000) (010) \rightarrow (110011) (011) \rightarrow (100101) (110) \rightarrow (010110)
 (001) \rightarrow (001111) (100) \rightarrow (111100) (101) \rightarrow (011001) (111) \rightarrow (101010)

So, if we are able to make a list of six questions that for every three-bit state of the chests gives us a corresponding six-bit answer vector, we will

be able to reconstruct the correct answer vector even if the parrot lies when answering one of the questions.

The questions can easily be constructed by using logical operations. Let us take a look at the first bit of all answer vectors. It is equal to one only in the third, fourth, fifth, and eighth vectors, which correspond to chest states (010) , (100) , (011) , (111) . So we can formulate the first question as follows:

“Is it true that there are treasures **only in the second chest** or **only in the first chest** or **only in the second and third chests** or **in all chests**?”

The truthful answers to these questions will give us the correct first bit of an answer vector. By constructing the other five questions similarly we can create a mapping as described above and will be able to decode six answers into a three-bit chest state even if one answer is not honest.

Originally, it was intended that the list of six questions should be presented before asking any of them. So, all questions are predetermined and do not depend on answers to other questions. But since this was not explicitly stated in the description of the problem, this condition was not required. Correct and full solutions to the problem were given by seven participants, most of them not using such a coding-theory approach and presenting interesting questioning strategies for the problem. The best solutions were presented by Alexander Grebennikov (Presidential PML 239, St. Petersburg, Russia) and by Ivan Baksheev (Gymnasium 6, Novosibirsk, Russia).

Problem: “Treasure chests”

Formulation

We have seven closed chests. Some of them contain treasure (diamonds, gold coins, and bitcoins), but we do not know which ones. A parrot knows which chests contain treasure and which do not; he agrees to answer questions with “yes” or “no.” He may possibly lie in his answers but not more than twice. List fifteen questions such that it is possible to deduce from the parrot’s answers which chests contain treasure and which do not.

Solution

See solution to the problem “Chests with treasure” for the idea of the solution. In this problem we need to find a code that maps all binary vectors of length seven (chest states) to binary vectors of length fifteen and can correct up to two errors. If such a code is found, then we can easily construct questions using the technique described in the solution for the “Chests with treasure” problem.

It is sufficient to find a $[15; 7; 5]$ -code, where “5” is the minimal distance of the code (since a code with the minimal distance five or greater can correct two errors). It is known that there exist Bose-Chaudhuri-Hocquenghem (BCH) codes with these exact parameters (for example, see MacWilliams and Sloane 1977). Using one of those codes we can construct fifteen questions for the parrot that will allow us to decode chest states from answers.

Full and correct solutions for the problem were proposed by twelve university students and professionals. All of them used error-correction codes.

Problem: “A numerical rebus”

Formulation

Buratino keeps his golden key in a safe that is locked with a numerical password. For secure storage of the password he replaced some digits in the password with letters (in such a way that different letters substitute different digits). After replacement Buratino got the password **NSUCRYPTO17**. Alice the Fox found out that

- the number **NSUCRYPTO** is divisible by all integers n , where $n < 17$, and
- the remainder of **NSU** – **CRY** is divisible by 7.

Can she find the password?

Remarks. Here we denote $\overline{ABC\dots}$ (see remarks to the problem “PIN code”) by **ABC...**

Solution

The main idea of the solution is to apply necessary divisibility rules that would allow us to reduce the exhaustive search for the original password. Let us describe the main steps of the solution.

1. Since the number **NSUCRYPTO** is divisible by all integers $1, 2, \dots, 16$, it is divisible by 720720. Thus, **O** is equal to 0.
2. Since **NSU** – **CRY** is divisible by 7, **PTO** is also divisible by 7.
3. Since **NSUCRYPTO** is divisible by 8, **PTO** is divisible by 8. Thus, we have that **PTO** is equal to 280, 560, or 840.
4. Since **O** is equal to 0 and **NSUCRYPTO** is divisible by 9, there is no digit 9 in this number.
5. Then we check all the possible variants while keeping in mind divisibility rules for 11 and 13. Finally, we find the unique number that is 376215840. So, the password is 37621584017.

The best solution was proposed by Andrei Razvan (“Fratii Buzesti” National College, Craiova, Romania). He implemented steps 1–3 and searched through the remaining possible numbers using a computer. While many proposed solutions used exhaustive searches, many students tried to find this number theoretically and got partial results.

Problem: “Timing attack”

Formulation

Anton invented a cipher machine that can automatically encrypt messages consisting of English letters. Each letter corresponds to a number from 1 to 26 by alphabetical order (1 is for A, 2 is for B, ... , 26 is for Z). The machine encrypts messages letter by letter. It encrypts one letter as follows:

1. If the letter belongs to the special secret set of letters, the machine does not encrypt it, adds the original letter to the ciphertext, and does not go to step 2; otherwise it goes to step 2.
2. According to the secret rule, it replaces the current letter with number k by a letter with number ℓ , where ℓ has the same remainder of division by 7, and adds this new letter to the ciphertext.

Anton’s classmate Evgeny is interested in different kinds of cryptanalysis that use some physical information about the encryption process. He measured the amount of time that is required for each letter encryption by Anton’s cipher machine and found out that a timing attack can be applied to it!

He captured the ciphertext that Anton sent to his friend and was able to read the message using the information from his measurements. Could you also decrypt the ciphertext

Tois kevy is fhye tvvu xust hgvtoed iyife ngfbey!

Wvat ka rvn knvw owvnt it?

if you know how much time encryption of each message letter took (Figure 1)?

Solution

First, one can notice that according to the diagram (Figure 1) an encryption of each letter took one or two seconds. This leads to an idea that the encryption process takes one second for letters from the secret group (only step 1 is needed) and takes two seconds otherwise (two steps of encryption). Thus, one can easily find letters that belong the secret group and partially read the message:

T_is_e__ is __e t__ __st __t_e__i_e __e_!

W__t __ __ k__w ____t it?

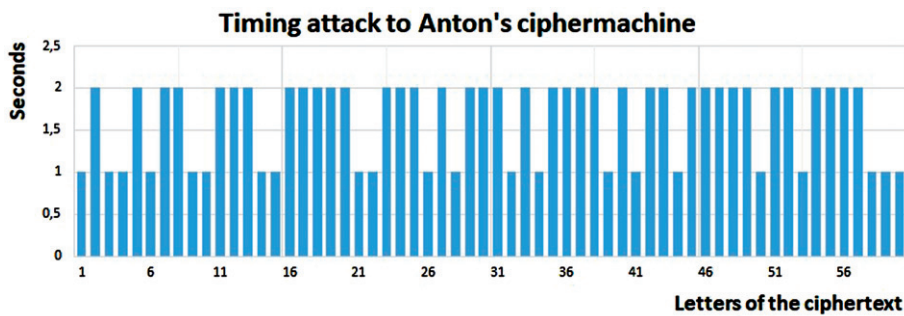


Figure 1. Time consumption of the message encryption.

To read the whole message we just need to write all possible replacements for each of the empty positions according to the step 2 rule to find the appropriate English words. Note that the second sentence “What do you know about it?” can be correctly guessed at the first glance! And many participants mentioned this observation. Let us read the first sentence:

T^a_o ^his k^d e^a h^k i s f^m a^h d^k e t^a ^h h^a gⁿ c^j gⁿ
v ^y v y t v y v v u x u

s t ^a hⁿ g^h a^h t^h e k^r bⁱ d^k i f^m e gⁿ gⁿ f^m ^b i^e k^r!
v u v v y w y t u t w y

Thus, the message is “This year is more than just another prime number! What do you know about it?” Really, the number 2017 has magnificent properties (see Wei 2018). Almost all participants solved the problem and found the correct answer. The best ones were given by school students Borislav Kirilov (FPMG, Sofia, Bulgaria) and Vladimir Schavelev (SESC NSU, Novosibirsk, Russia) and by university students Igor Antonov (Ural State University of Railway Transport) and Kristina Zhuchenko (Demidov Yaroslavl State University).

Problem: “A music lover”

Formulation

As usual Alex listens to music on the way to university. He chooses it by applying one secret code to the second one in his mind (Figure 2). Could you understand what music he is listening to right now?

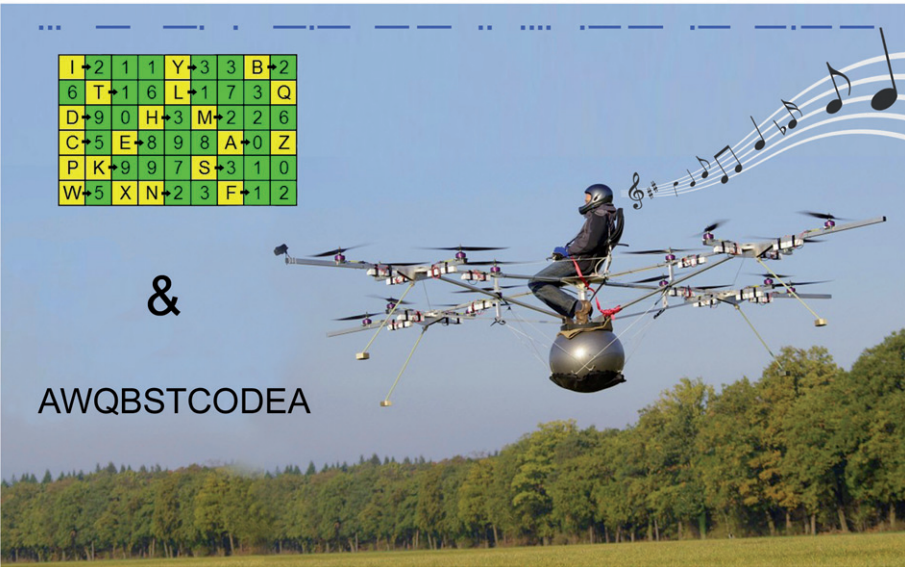


Figure 2. Illustration for the problem “A music lover.”

Remarks.

- 1. You should invent a way to apply one code to another.
- 2. Some arithmetic operations also can be used.

Solution

Let us look at Figure 2. The first natural step in solving the problem is to decode the Morse Code at the top of the picture: STNEKMIHWAY. The decoded string consists of 11 letters as well as the string AWQBSTCODEA under the table. It brings to the mind that we should somehow apply one string to another using the information from the table. Note that almost every letter in the table has an arrow pointing to a sequence of numbers, and among the letters having arrows there are all the letters of the decoded string STNEKMIHWAY:

S → 3 1 0	T → 1 6	N → 2 3	E → 8 9 8	K → 9 9 7	M → 2 2 6
I → 2 1 1	H → 3	W → 5	A → 0	Y → 3 3	

Let us sum the numbers for each letter above. That is, we need to calculate $3 + 1 + 0$, $1 + 6$, and so on. Thus, we get the string of eleven integers, 4, 7, 5, 25, 25, 10, 4, 3, 5, 0, 6. Finally, we can apply this string to the AWQBSTCODEA in the following way: each letter is cyclically shifted right in the alphabet by the corresponding number of positions from the integer string. We get

letter	A	W	Q	B	S	T	C	O	D	E	A
shift	4	7	5	25	25	10	4	3	5	0	6
result	E	D	V	A	R	D	G	R	I	E	G

Thus, we conclude that Alex listens to the music of the great Norwegian composer Edvard Grieg.

The problem appeared to be difficult for the first round: it was completely solved by only three participants. Nevertheless, twelve teams solved the problem during the second round.

Problem: “The shortest addition chain”

Formulation

In many cryptographic systems we need to calculate the value $B = A^c \bmod p$, where A is an integer, $1 \leq A \leq p-1$, c is an arbitrary positive integer, and p is a large prime number. One possible way of reducing the computational load of calculation is to minimize the total number of multiplications required to compute the exponentiation. Since the exponent in the equation is additive, the problem of computing powers of the base element A can also be formulated as an addition calculation for which addition chains are used.

An addition chain for an integer n is a sequence of positive integers $a_0 = 1, a_1, \dots, a_{r-1}, a_r = n$, where r is a positive integer (called the length of the addition chain), and the following relation holds for all i , $1 \leq i \leq r$: $a_i = a_j + a_k$ for some k, j such that $k \leq j < i$.

Find an addition chain of length as small as possible for the value 81, present it as a list of values, and mathematically prove that it cannot be shorter!

An example: For the value 15 the shortest additional chain has length 5 and its list of values is 1, 2, 3, 6, 12, 15. So, to optimally calculate $B = A^{15} \bmod p$, one can use just five multiplications: $A^2 = A \cdot A \bmod p$, $A^3 = A^2 \cdot A \bmod p$, $A^6 = A^3 \cdot A^3 \bmod p$, $A^{12} = A^6 \cdot A^6 \bmod p$, and $A^{15} = A^{12} \cdot A^3 \bmod p$.

Solution

It is easy to construct an addition chain of length 8 for value 81. For example,

$$1, 2, 4, 8, 16, 32, 64, 80, 81.$$

There are different ways of proving that a chain of length 7 or shorter does not exist. One way is to write a computer program that will construct all possible addition chains of length 1, ..., 7 and show that none of them contain 81. We will provide a theoretical proof.

Let us have an addition chain $1 = a_0, a_1, a_2, \dots, a_r$ of length r . A trivial observation is that a_k cannot be greater than 2^k for any k . Thus, an addition chain of length 6 or shorter cannot exist for the number 81 because

$81 > 64 = 2^6$. So, we have to prove that an addition chain of length 7 is not possible either. In order to do that, we prove the following lemma:

Lemma. Let $1 = a_0, a_1, a_2, \dots, a_r$ be an addition chain of length r . Assume that $2^{r-1} < a_r \leq 2^r$. Then, $a_r = 2^{r-1} + 2^s$ for some $0 \leq s \leq r-1$.

Proof. Let us prove the lemma by induction on r . For $r=1$, there is only one addition chain $(1, 2)$ that satisfies the condition of the lemma. Assume that for all addition chains $a_0, a_1, a_2, \dots, a_t$ of length $t < r$, such that $2^{t-1} < a_t \leq 2^t$, it holds $a_t = 2^{t-1} + 2^s$, where $0 \leq s \leq t-1$.

Let $1 = a_0, a_1, \dots, a_{t+1}$ be an addition chain of length $t+1$ such that $2^t < a_{t+1} \leq 2^{t+1}$. By definition of an addition chain, $a_{t+1} = a_n + a_m$ for some indices $n, m \leq t$. If both a_n, a_m are not greater than 2^{t-1} , then $a_{t+1} \leq 2^{t-1} + 2^{t-1} = 2^t$; that is a contradiction.

Therefore, without loss of generality, $a_n > 2^{t-1}$. But that also means $n > t-1$. So, $n = t$ and the chain a_0, a_1, \dots, a_t satisfy the induction hypothesis. Therefore, $a_n = a_t = 2^{t-1} + 2^s$ for some $s \leq t-1$. Substituting a_n into the expression for a_{t+1} , we obtain

$$a_{t+1} = 2^{t-1} + 2^s + a_m$$

for some $m \leq t$ and $s \leq t-1$. Now consider several cases:

- $s = t-1$. In this case, $a_n = a_t = 2^t$, which forces all a_i to be equal to 2^i . So, $a_{t+1} = 2^t + 2^m$ and the induction step is proven.
- $m = t$. In this case, $a_{t+1} = 2a_t = 2^t + 2^{s+1}$ and the induction step is proven.
- $s < t-1, m = t-1, a_m \leq 2^{t-2}$. In this case, $a_{t+1} = 2^{t-1} + 2^s + a_m \leq 2^{t-1} + 2^{t-2} + 2^{t-2} = 2^t$; that is a contradiction.
- $s < t-1, m = t-1, a_m > 2^{t-2}$. Then, a_0, a_1, \dots, a_{t-1} satisfies the induction hypothesis and $a_m = 2^{t-2} + 2^q, q \leq t-2$. Thus, $a_{t+1} = 2^{t-1} + 2^s + 2^{t-2} + 2^q$.
 - If $q = t-2$, then $a_{t+1} = 2^t + 2^s$ and the induction step is proven.
 - If $s = t-2$, then $a_{t+1} = 2^t + 2^q$ and the induction step is proven.
 - If $q < t-2$ and $s < t-2$, then $a^{t+1} \leq 2^t$, which is a contradiction.
- $s < t-1, m < t-1$. Then $a_m \leq 2^{t-2}$ and $a_{t+1} \leq 2^{t-1} + 2^{t-2} + 2^{t-2} = 2^t$, which is a contradiction.

We considered all cases and checked that some of them contradict the lemma assumption while others lead to the proven induction step. Thus the lemma holds.

Now assume that there is an addition chain of length 7 for 81. Because $81 > 64 = 2^6$ such a chain would satisfy the condition of the lemma. Then, $81 = a_7 = 64 + 2^s$ for some $s \leq 6$. This is impossible. So, the shortest addition chain for 81 has length 8.

This is a rather general solution that also provides an interesting fact about addition chains. Just four school students completely solved the problem; the best solution was provided by Alexander Dorokhin (Presidential PML 239, St. Petersburg, Russia). Usually, proofs of impossibility of a chain of length 7 were given in a more straightforward manner, which involved checking possible strategies of getting to number 81 in 7 steps and showing that no matter how we add numbers, we will not be able to get 81.

Problem: “An infinite set of collisions”

Formulation

Bob is very interested in blockchain technology, so he decided to create his own system. He started with the construction of a hash function. His first idea for a hash function was the function H with a hash value of length 16.

It works as follows.

- Let $u_1, u_2, \dots, u_n \in \mathbb{F}_2$ be a data representation; n is arbitrary.
- Bob calculates $z^0, \dots, z^n \in \mathbb{F}_2^{32}$, $z^0 = (0, \dots, 0)$, and z^{i+1} is obtained from z^i in the following way:

$$z' = \begin{cases} (z_1^i, z_2^i, \dots, z_{16}^i, z_1^i \oplus z_{17}^i, z_2^i \oplus z_{18}^i, \dots, z_{16}^i \oplus z_{32}^i) & \text{if } u_i = 1, \\ (z_1^i \oplus z_{17}^i, z_2^i \oplus z_{18}^i, \dots, z_{16}^i \oplus z_{32}^i, z_{17}^i, z_{18}^i, \dots, z_{32}^i) & \text{if } u_i = 0, \end{cases}$$

$$z'' = \begin{cases} z' & \text{if } u_i \neq z_{32}^i, \\ (z_1' \oplus 1, z_2' \oplus 1, \dots, z_{32}' \oplus 1) & \text{if } u_i = z_{32}^i, \end{cases}$$

$$z^{i+1} = (z_2'', z_3'', \dots, z_{32}'', u_i).$$

- Finally, $H(u_1, \dots, u_n) = (z_1^n \oplus z_{17}^n, z_2^n \oplus z_{18}^n, \dots, z_{16}^n \oplus z_{32}^n)$.

But then Bob found out that his hash function is weak for use in cryptographic applications. Prove that Bob was right by constructing an infinite set $C \subset \bigcup_{n=1}^{\infty} \mathbb{F}_2^n$ such that all elements of C have the same hash value H .

An example: Let us calculate $H(0, 1, 0)$. We have

$$z^1 = (\underbrace{1, 1, \dots, 1}_{31}, 0), z^2 = (\underbrace{0, \dots, 0}_{15}, \underbrace{1, \dots, 1}_{15}, 0, 1),$$

$$z^3 = (\underbrace{1, \dots, 1}_{13}, 0, 0, \underbrace{1, \dots, 1}_{14}, 0, 1, 0).$$

$$\text{Thus, } H(0, 1, 0) = (\underbrace{0, \dots, 0}_{14}, 1, 1).$$

Solution

Here we provide the solution proposed by Alexey Udovenko (University of Luxembourg). It consists of a theoretical proof and a simple example. Exactly the same idea was suggested by the program committee.

A theoretical proof: Let us consider an arbitrary infinite sequence $u = u_1, u_2, u_3, \dots$ and the following hash values:

$$H(u_1), H(u_1, u_2), H(u_1, u_2, u_3), \dots$$

To obtain an infinite set of collisions, it is enough to find some ℓ, m such that $z^\ell = z^m$. Then we reconstruct the sequence in the following way: $u_k = u_{k-|m-\ell|}$ starting with $k = \max\{\ell, m\} + 1$. By this method we obtain a cycle in the sequence of the states z^1, z^2, z^3, \dots , since each state z^i uniquely defines $H(u_1, \dots, u_i)$ and the next state z^{i+1} in conjunction with u_{i+1} . The cycle length divides $2^{32}!$ because it is between 1 and $|\mathbb{F}_2^{32}| = 2^{32}$. The initial state z_0 may not belong to the cycle, but after 2^{32} steps $z^{2^{32}}$ definitely belongs to the cycle. It means that

$$H(u_1, \dots, u_{2^{32}}) = H(u_1, \dots, u_{2^{32}+1 \cdot 2^{32}!}) = H(u_1, \dots, u_{2^{32}+2 \cdot 2^{32}!}) = \dots$$

An example: Let us consider the zero sequence $u = 0, 0, 0, \dots$. In this case,

$$z^{31} = (0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0)$$

and $z^{32} = z^{31}$. Thus, $H(u_1, \dots, u_k) = (1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0)$ for any $k \geq 31$.

The problem was completely solved by twelve university students and professionals.

Problem: "One more parameter"**Formulation**

There are several parameters of block ciphers in cryptanalysis that are used to measure diffusion strength. In this problem we study properties of one of them.

Let n, m be positive integers. Let $a = (a_1, \dots, a_m)$ be a vector with coordinates a_i taken from the finite field \mathbb{F}_2 . Denote the number of nonzero coordinates a_i , $i = 1, \dots, m$ by $\text{wt}(a)$ and call this number the weight of the vector a . The inner product of $a = (a_1, \dots, a_m)$ and $b = (b_1, \dots, b_m)$ in \mathbb{F}_2^m is defined as $a \cdot b = a_1 b_1 \oplus \dots \oplus a_m b_m$. For a Boolean function $f : \mathbb{F}_2^m \rightarrow \mathbb{F}_2$, we define the function weight, wt , as follows: $\text{wt}(f) = |\{a \in \mathbb{F}_2^m \mid f(a) = 1\}|$.

The special parameter Q of a vectored Boolean function $\varphi : \mathbb{F}_2^m \rightarrow \mathbb{F}_2^m$ is defined to be

$$Q(\varphi) = \min_{a,b,b \neq 0, \text{wt}(a \cdot x \oplus b \cdot \varphi(x)) \neq 2^{m-1}} \{\text{wt}(a) + \text{wt}(b)\}.$$

- Rewrite (simplify) the definition of $Q(\varphi)$ when the function φ is linear (recall that a function ℓ is linear if $\ell(x \oplus y) = \ell(x) \oplus \ell(y)$ for any x, y).
- Rewrite the definition of $Q(\varphi)$ in terms of linear codes when the linear function φ is given by an $m \times m$ matrix M over \mathbb{F}_2 (i.e., $\varphi(x) = Mx$).
- Find the tight upper bound for $Q(\varphi)$ as a function of m .
- Can you give an example of the function φ with the maximal possible value of Q ?

Solution

A special parameter Q considered in the problem is called the “linear branch number” of a transformation (Daemen and Rijmen 2002). This problem is a linear cryptanalysis equivalent to the problem “A special parameter” of NSUCRYPTO-2014 (Agievich et al. 2015), where the differential branch number was discussed.

Let φ be a vectored Boolean function $\mathbb{F}_2^m \rightarrow \mathbb{F}_2^m$.

- If φ is a linear function, then the Boolean function $a \cdot x \oplus b \cdot \varphi(x)$ is also linear for any vectors $a, b \in \mathbb{F}_2^m$. Hence, the condition $\text{wt}(a \cdot x \oplus b \cdot \varphi(x)) \neq 2^{m-1}$ is equivalent to $a \cdot x \oplus b \cdot \varphi(x) = 0$ for all $x \in \mathbb{F}_2^m$. Thus, for the considered case we have the definition

$$Q(\varphi) = \min_{a,b,b \neq 0, a \cdot x \oplus b \cdot \varphi(x) \equiv 0} \{\text{wt}(a) + \text{wt}(b)\}.$$

- Let us consider vectors as columns. In the case when $\varphi(x) = Mx$ for some $m \times m$ matrix M over the field \mathbb{F}_2 , we can rewrite $a \cdot x \oplus b \cdot \varphi(x) = (a \oplus M^T b) \cdot x$. Then, $(a \oplus M^T b) \cdot x \equiv 0$ implies $a \oplus M^T b = 0$ or $Hc = 0$, where $H = (I|M^T)$ is a $m \times 2m$ matrix, I is the identity $m \times m$ matrix, and $c = (a, b)$ denotes the concatenation of vectors a and b of length $2m$. Note that $b = 0$ and $a = M^T b$ imply $a = 0$. So, $b \neq 0$ is equivalent to $c \neq 0$. Thus,

$$Q(\varphi) = \min_{c \neq 0, Hc=0} \{\text{wt}(c)\} = \text{dist}(C),$$

where C is the linear code of length $2m$ and dimension m with a parity-check matrix H and $\text{dist}(C)$ denotes the distance of code C .

- Here we would like to apologize to the participants since the formulation of the problem was not correctly stated. If we consider a mapping $\varphi : \mathbb{F}_2^m \rightarrow \mathbb{F}_2^m$ instead of $\varphi : \mathbb{F}_2^m \rightarrow \mathbb{F}_2^n$, then one can easily find the bound $Q(\varphi) \leq m + 1$, and this bound is tight for various parameters n

and m . Indeed, there exist maximal distance separable codes with parameters $[2m, m, m + 1]$ over \mathbb{F}_2^n (for example, Reed-Solomon codes).

At the same time, for $\varphi : \mathbb{F}_2^m \rightarrow \mathbb{F}_2^m$ as it was given in the problem, the bound $Q(\varphi) \leq m + 1$ can be achieved only when $m = 1$. So, we cannot say that this bound is tight for various m . Therefore this bound cannot be considered as a correct answer.

To be honest, we cannot say the correct answer to the problem, so we may assume that this problem is also one of the open problems of the Olympiad. What was surprising and very pleasant for us is that several teams found nontrivial bounds for general and linear cases. But unfortunately, they could not say if these bound were tight. We would like to shortly present the main results of the participants.

The linear case: $Q(\varphi) \leq (2m + 4)/3$. This bound was found by Irina Slonkina (National Research Nuclear University MEPhI).

Let us consider any $m \times m$ matrix S over the field \mathbb{F}_2 and a linear function $\varphi_s(x) = Sx$. So,

$$Q(\varphi_s) = \min_{b \neq 0} \{ \text{wt}(bS) + \text{wt}(b) \}.$$

It is clear that for any $i, j \in \{1, 2, \dots, m\}, i \neq j$, the following bounds hold:

$$Q(\varphi_s) \leq \text{wt}(S_i) + 1 \text{ and } Q(\varphi_s) \leq \text{wt}(S_i \oplus S_j) + 2,$$

where $S_i, S_j \in \mathbb{F}_2^m$ are i th and j th rows of the matrix S . Then it holds that

$$\begin{aligned} Q(\varphi_s) - 2 &\leq \text{wt}(S_i \oplus S_j) \leq 2m - \text{wt}(S_i) - \text{wt}(S_j) \leq \\ &\leq 2m - (Q(\varphi_s) - 1) - (Q(\varphi_s) - 1) = 2m - 2Q(\varphi_s) + 2, \end{aligned}$$

where the second bound follows from the inequality $\text{wt}(u \oplus v) \leq 2m - \text{wt}(u) - \text{wt}(v)$ that holds for any $u, v \in \mathbb{F}_2^m$. Thus, for the function φ_s we have the bound $Q(\varphi_s) \leq (2m + 4)/3$.

The general case: If $m \geq 2$, then $Q(\varphi) \leq m$.

This bound was found by the Alexey Miloserdov, Saveliy Skresanov, and Nikita Odinokih team (Novosibirsk State University) and by the Kristina Geut and Sergey Titov team (Ural State University of Railway Transport). Here we present the solution of the first team.

Let $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$. The Walsh transform of f is defined as $W_f(y) = \sum_{x \in \mathbb{F}_2^n} (-1)^{y \cdot x \oplus f(x)}$, $y \in \mathbb{F}_2^n$. The function f is uniquely defined by its Walsh coefficients since the following equality holds:

$$(-1)^{f(x)} = \frac{1}{2^n} \sum_{y \in \mathbb{F}_2^n} W_f(y) (-1)^{y \cdot x}, x \in \mathbb{F}_2^n.$$

It is also well known that Parseval's equality, $\sum_{y \in \mathbb{F}_2^n} W_f^2(y) = 2^{2n}$, holds for any Boolean function f .

Proposition: Let $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$. Suppose that for every $a \in \mathbb{F}_2^n$ such that $0 \leq \text{wt}(a) \leq n-1$ it holds that $\text{wt}(a \cdot x \oplus f(x)) = 2^{n-1}$. Then, $f(x) = x_1 \oplus x_2 \oplus \dots \oplus x_n \oplus c$ for some $c \in \mathbb{F}_2$.

Proof. Since a function $a \cdot x \oplus f(x)$ is balanced if $W_f(a) = 0$, then by Parseval's equality and the assumption of the proposition we have $W_f(a) = 0$ for all $a \in \mathbb{F}_2^n$ such that $0 \leq \text{wt}(a) \leq n-1$ and $|W_f(1)| = 2^n$, where $1 = (1, 1, \dots, 1) \in \mathbb{F}_2^n$. In this case it holds that $(-1)^{f(x)} = \frac{\pm 2^n}{2^n} (-1)^{1 \cdot x}$, $x \in \mathbb{F}_2^n$; that is, $f(x) = x_1 \oplus x_2 \oplus \dots \oplus x_n \oplus c$, for some $c \in \mathbb{F}_2$.

Corollary. $Q_{\max}(m) \leq m$ for $m \geq 2$, where $Q_{\max}(m) = \max_{\varphi: \mathbb{F}_2^m \rightarrow \mathbb{F}_2^m} Q(\varphi)$.

Proof. Denote $b_1 = (1, 0, 0, \dots, 0) \in \mathbb{F}_2^m$, $b_2 = (0, 1, 0, \dots, 0) \in \mathbb{F}_2^m$. Assume that there exists some φ such that $Q(\varphi) > m$. It implies $\text{wt}(a \cdot x \oplus b_i \cdot \varphi(x)) = 2^{m-1}$, $i = 1, 2$ for any constant $a \in \mathbb{F}_2^m$ such that $0 \leq \text{wt}(a) \leq m-1$. Then, by the proposition it holds $b_i \cdot \varphi(x) = 1 \cdot x \oplus c_i$, for some $c_i \in \mathbb{F}_2$, $i = 1, 2$. Thus, the sum modulo 2 of the first and second coordinate functions of φ is a constant function. Hence, $\text{wt}(b \cdot \varphi(x)) \in \{0, 2^m\}$, where $b = (1, 1, 0, \dots, 0) \in \mathbb{F}_2^m$. But then we have $Q(\varphi) \leq 2$; that is a contradiction.

Problem: "Scientists"

Formulation

Alice and Bob, two young cryptographers and very curious students, studied different cryptosystems and attacks on them. At the same time they were very interested in biographies of famous scientists and found out one interesting property that can be used in cryptosystems. They chose three pairs of scientists:

Charles Darwin and Michael Faraday,
Werner Heisenberg and Johannes Kepler,
Hans Christian Orsted and Mikhail Lomonosov.

Alice and Bob chose a cryptosystem and an attack they would like to study. They constructed three sets of parameters for the cryptosystem, one set according to each pair of scientists. Then Alice chose a phrase consisting of 18 English letters (spaces were omitted) and divided it into three parts of six letters each. She represented each part as a hexadecimal

number using ASCII code. Alice encrypted the first part by the cryptosystem for each set of parameters, then did the same for the second and the third parts. Finally, Alice got the following three groups of three cipher texts (in hexadecimal notation).

	Part 1	Part 2	Part 3
Set of parameters 1	2512 1F5A 0079	B494 222D 3E1C	275E B751 4FDB
Set of parameters 2	3D0D 6812 0443	5111 5BFD 9398	0815 6223 2698
Set of parameters 3	1EDC 4856 8CE2	9C18 2A32 B9AB	9A1C AD5C 25D7

She asked Bob to decrypt it using the attack! Bob successfully read the secret phrase. Could you

- find the property like Alice and Bob,
- understand what is the cryptosystem and the attack chosen,
- decrypt the cipher text by applying this attack?

*What word should be added at the beginning of the decrypted text according to the famous words of Mikhail Lomonosov?

Solution

The problem is related to the RSA cryptosystem (Rivest, Shamir, and Adleman 1978) and the broadcast attack is related to Hastad 1988. Given pairs of famous scientists, some participants made a correct guess that each of the pairs is linked to some prime numbers P , Q , and the RSA modulus $N = PQ$. But what is the way to obtain the prime numbers? Success in their searches depends only on their intuition. Writing down the birthdate of each of the scientists in the form DDMMYYYY, one can notice that all these seven- or eight-decimal numbers are prime:

$$P_1 = 12021809 \text{ and } Q_1 = 22091791,$$

$$P_2 = 5121901 \text{ and } Q_2 = 27121571,$$

$$P_3 = 14081777 \text{ and } Q_3 = 19111711.$$

This is the property Alice and Bob found in the biographies of the scientists.

Since Alice encrypted each part of the original text thrice, in order to decrypt it we can try to apply Hastad's broadcast attack on RSA as it is described in Boneh 1999. Three pairs of parameters should indicate that $e=3$ was chosen as the public exponent. This is not a good decision on the part of Alice and Bob. A valid public exponent must be coprime with $\varphi(N)$ because that makes it possible to compute the private exponent, whereas three divides either $\varphi(N_1)$, $\varphi(N_2)$ or $\varphi(N_3)$. Two solutions noted this weird choice and both of them are marked as the best solutions.

Nevertheless, it is still reasonable to use the Chinese remainder theorem and take the cubic root. In other words, by applying Hastad's broadcast attack we obtain three parts of the message encrypted. Converting them back to ASCII characters we get the plaintext PUTSTHEMINDINORDER, which means PUTS THE MIND IN ORDER. This is a part of the famous phrase by a distinguished Russian scientist Mikhail Lomonosov, who said that "Mathematics should be studied because it puts the mind in order." Consequently, the first word of the quote is "mathematics".

The participants presented two comprehensive solutions at the first round and seven at the second round. One more solution turned out to be almost complete: authors pointed to a wrong word as the beginning of Lomonosov's statement. The best solutions were proposed by the team of Daniel Malinowski and Michal Kowalczyk (University of Warsaw, Dragon Sector) and by the team of Alexey Ripinen, Oleg Smirnov, and Peter Razumovsky (Saratov State University).

Problem: "Masking"

Formulation

It is known that there are attacks on cryptosystems that use information obtained from the physical implementation of a cryptosystem; for example, timing information, power consumption, electromagnetic leaks, or even sound. To protect cryptosystems from such attacks cryptographers can use a countermeasure known as masking.

Correlation-immune Boolean functions can reduce the masking cost. Therefore, we need to search for Boolean functions that satisfy the following conditions: they should have a small Hamming weight, for implementation reasons, and a high correlation immunity to resist an attacker with multiple probes.

Let f be a nonconstant Boolean function in 12 variables of correlation immunity equal to 6.

- What is the lowest possible Hamming weight k of f ?
- Give an example of such a function f with Hamming weight k .

Remarks.

1. Hamming weight $\text{wt}(f)$ of a Boolean function f in n variables is the number of vectors $x \in \mathbb{F}_2^n$ such that $f(x) = 1$.
2. A Boolean function f in n variables is called correlation immune of order t , where t is an integer such that $1 \leq t \leq n$, if $\text{wt}(f_{i_1, \dots, i_t}^{a_1, \dots, a_t}) = \text{wt}(f)/2^t$ for any set of indexes $1 \leq i_1 < \dots < i_t \leq n$ and any set of values $a_1, \dots, a_t \in \mathbb{F}_2$. Here $f_{i_1, \dots, i_t}^{a_1, \dots, a_t}$ denotes the subfunction of f in $n - t$

variables that is obtained from $f(x_1, \dots, x_n)$ by fixing each variable x_{i_k} by the value a_k , $1 \leq k \leq t$.

Solution

First we should note that this problem contains open questions in general. We considered the participants' solutions as correct if they were as deep as solutions known to the Olympiad Program Committee. More precisely, we expected from the participants Boolean functions in 12 variables of weight 1,024 that are correlation immune of order 6.

Let f be a nonconstant Boolean function in n variables of correlation immunity t and of Hamming weight k . The known open problem is to find such a function f having as low as possible Hamming weight for various n and t . The problem questions were investigated in Bhasin, Carlet, and Guilley 2013 as well as Picek et al. 2015, where minimal Hamming weight $k = 1,024$ of f for $n = 12$ and $t = 6$ was found using heuristics (more precisely, evolutionary algorithms). Theoretically, k can be lower than 1,024, but any example of such a function is unknown. It can be equal to any value of the form $k = 64\ell$ greater than or equal to 768 according to the results of orthogonal arrays (Hedayat, Sloane, and Stufken 1999). It is known that the elements of the support of f form the rows of an orthogonal array with parameters $(k, n, 2, t)$ (recall that $x \in \mathbb{F}_2^n$ belongs to the support of f if $f(x) = 1$).

We present several constructions of f proposed by the participants.

1. The first compact example was obtained by the team of Maxim Plushkin, Ivan Lozinskiy, and Azamat Miftakhov (Lomonosov Moscow State University). They found the following function:

$$f(x_1, \dots, x_{12}) = (x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus 1) \cdot \\ (x_6 \oplus x_7 \oplus x_8 \oplus x_9 \oplus x_{10} \oplus x_{11} \oplus x_{12} \oplus 1).$$

Note that the team studied the problem for a small number of variables n up to 14 and correlation-immunity $t = n/2$. For example, they found a function in 10 variables with the Hamming weight $k = 256$ constructed similarly to the case of 12 variables:

$$f(x_1, \dots, x_{10}) = (x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus 1) \cdot \\ (x_5 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_9 \oplus x_{10} \oplus 1).$$

And as proved in Bhasin, Carlet, and Guilley (2013) this weight cannot be lower.

2. Another solution was found by Alexey Udovenko (University of Luxembourg) in the following way:

$$f(x_1, \dots, x_{12}) = s_1 \oplus s_1 s_2 \oplus s_1 s_3 \oplus s_2 s_3,$$

where $s_1 = x_1 \oplus x_2 \oplus x_3 \oplus x_4$, $s_2 = x_5 \oplus x_6 \oplus x_7 \oplus x_8$, $s_3 = x_9 \oplus x_{10} \oplus x_{11} \oplus x_{12}$. Alexey also mentioned that in the case of the quadratic function in 12 variables the Hamming weight cannot be less than 1,024 (Proposition 1.9 of Canteaut 2016). Note that he concentrated his search on quadratic functions whose graphs of quadratic terms have multiple automorphisms. This idea was supported by studying the graph of quadratic terms of a function f with parameters $n=6$, $t=3$, and $k=16$. Alexey computationally proved that, in this case, 16 is the minimal Hamming weight of f .

3. The third interesting example was proposed by Anna Taranenko (Sobolev Institute of Mathematics) and can be described as follows:

$$\begin{aligned} f(x_1, \dots, x_{12}) = 1 \iff & \varphi(x_1, x_2, x_3) + \varphi(x_4, x_5, x_6) + \varphi(x_7, x_8, x_9) \\ & + \varphi(x_{10}, x_{11}, x_{12}) = 0, \end{aligned}$$

where $+$ denotes the addition in \mathbb{Z}_4 and φ takes the following values: $\varphi(0, 0, 0) = \varphi(1, 1, 1) = 0$; $\varphi(1, 0, 0) = \varphi(0, 1, 1) = 1$; $\varphi(0, 1, 0) = \varphi(1, 0, 1) = 2$; $\varphi(0, 0, 1) = \varphi(1, 1, 0) = 3$. Anna presented a mathematical proof that the function f is correlation immune of order 7 (and therefore 6) with the Hamming weight 1,024. She also mentioned that for a 12-variables function of correlation-immunity 7 the minimal Hamming weight is exactly 1,024 according to the Bierbrauer-Friedman inequality for parameters of orthogonal arrays (Bierbrauer 1995; Friedman 1992).

Overall only these three teams mentioned above made significant progress with this problem.

Problem: “TwinPeaks”

Formulation

On Bob’s smartphone there is a program that encrypts messages with the algorithm TwinPeaks. It works as follows:

1. It takes an input message P that is a hexadecimal string of length 32 and represents it as a binary word X of length 128.
2. Then X is divided into four 32-bits words a, b, c, d .

- Then six rounds of the following transformation are applied:

$$(a, b, c, d) \leftarrow (a + c + S(c + d), a + b + d + S(c + d), \\ a + c + d, b + d + S(c + d)),$$

where S is a secret permutation from \mathbb{F}_2^{32} to itself and $+$ denotes the coordinate-wise sum modulo 2.

- The word Y is obtained as a concatenation of a, b, c, d .
- Finally, Y is converted to the hexadecimal string C of length 32. The algorithm gives C as the cipher text for P .

Agent Cooper intercepted the cipher text

$C = 59A0D027D032B394A0A47A9ED19C98A8$

sent from Bob to Alice and decided to decrypt it.

In order to solve this problem Agent Cooper also captured Bob's smart-phone with the TwinPeaks algorithm. [Here it is](#). Now Cooper (and you too) can encrypt any messages with TwinPeaks but still cannot decrypt a single one.

Help Cooper to decrypt C .

Solution

Let F be the round transformation of TwinPeaks:

$$F(a, b, c, d) = (a + c + S(c + d), a + b + d + S(c + d), \\ a + c + d, b + d + S(c + d))$$

and

$$f(a, b, c, d) = (c + d, a + b + c, a + b, b + c + d).$$

If F transforms a message (a, b, c, d) to (a', b', c', d') , then G transforms a message $f(a, b, c, d)$ to $f(a', b', c', d')$, where G acts as follows:

$$G(a, b, c, d) = (b + S(a), c, d, a).$$

This conclusion can be extended to all six rounds. It will be convenient to consider a modification of TwinPeaks, where F is replaced by G . Indeed, one can encrypt a message (x_1, x_2, x_3, x_4) , where $x_i \in \mathbb{F}_2^{32}$, using the modified algorithm in the following way:

- Encrypt $f^{-1}(x_1, x_2, x_3, x_4) = (x_1 + x_3 + x_4, x_1 + x_4, x_2 + x_3, x_1 + x_2 + x_3)$ with TwinPeaks.
- Transform the encryption result using f .

Figure 3 illustrates the modified TwinPeaks. As one can see, plaintexts and cipher texts are linked with the following relations:

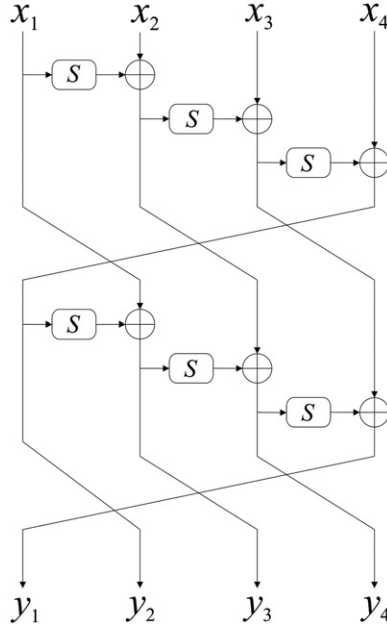


Figure 3. Modified TwinPeaks.

$$\begin{aligned}
 y_1 &= x_3 + S(x_2 + S(x_1)) + S(y_4), \\
 y_2 &= x_4 + S(x_3 + S(x_2 + S(x_1))), \\
 y_3 &= x_1 + S(y_2), \\
 y_4 &= x_2 + S(x_1) + S(y_3).
 \end{aligned}$$

Cooper can choose as x_1 any value u and get its representation in the form $u = v + S(w)$, where $v = y_3$ and $w = y_2$. Moreover, Cooper can represent w in the form $v' + S(w')$ and finally get a representation

$$u = v + S(v' + S(w')).$$

Suppose that Cooper wants to find x_1 . Then he represents

$$y_2 = v + S(v' + S(w'))$$

(two requests to TwinPeaks) and encrypts (w', v', v, y_3) (one request). The second word of the cipher text is

$$y_3 + S(v + S(v' + S(w'))) = y_3 + S(y_2) = x_1.$$

Since Cooper is able to find x_1 given (y_1, y_2, y_3, y_4) , he can calculate $S(u)$ for all u . Indeed, Cooper can choose $y_2 = u$ and an arbitrary y_1, y_3, y_4 . Then he finds x_1 and $S(u) = x_1 + y_3$. This can be completed using three requests.



Figure 4. Awards ceremony at Novosibirsk State University, December 2017.



Figure 5. The NSUCRYPTO winners of different years.

Using six requests Cooper can find $S(x_1) + S(y_3)$ and, hence, x_2 . Another six requests is enough to find $S(x_2 + S(x_1)) + S(y_4)$ and, hence, x_3 . Finally, using three request he finds $S(x_3 + S(x_2 + S(x_1)))$ and, hence, x_4 .

Thus, one needs $3 + 6 + 6 + 3 = 18$ requests to decrypt the message. The answer is 43ABECCAA53CB953F35239E79CC900EE.

Correct solutions were proposed by twenty teams of university students and professionals. All of them used different methods and techniques. We did not identify a single best solution, but we are pleased to note the participants' professionalism and creativity in solving the problem.

Problem: "An addition chain"

Formulation

In many cryptographic systems we need to calculate the value $B = A^c \bmod p$, where A is an integer, $1 \leq A \leq p-1$, c is an arbitrary positive integer, and p is a large prime number. One possible way of reducing the computational load of calculation is to minimize the total number of multiplications required to compute the exponentiation. Since the exponent in the equation is additive, the problem of computing powers of the base

element A can also be formulated as an addition calculation for which addition chains are used.

An addition chain for an integer n is a sequence of positive integers $a_0 = 1, a_1, \dots, a_{r-1}, a_r = n$, where r is a positive integer (called the length of the addition chain) and the following relation holds for all i , $1 \leq i \leq r$: $a_i = a_j + a_k$ for some k, j such that $k \leq j < i$.

Find an addition chain of length as small as possible for the value $2^{127} - 3$.

The solution should be submitted as a list of values occurring in the chain and a description of how you found the solution. An example of the shortest addition chain for the value 15 can be found in formulation of the problem “The shortest addition chain”.

Solution

We should note that the problem contains open questions in general. We considered the participants’ solutions as correct if they are as deep as solutions known to the Olympiad Program Committee. More precisely, if the participants could find an addition chain of length 136.

We would like to follow the solution proposed by the team of Alexey Miloserdov, Saveliy Skresanov, and Nikita Odinokih (Novosibirsk State University). Denote by $\ell(n)$ the length of the smallest addition chain for a number n . Let us first prove that $\ell(2^{127} - 3) \leq 136$ and present a chain of length 136. Then we will consider several lower bounds for $\ell(2^{127} - 3)$.

It is easy to see that

$$2^{127} - 3 = 4(2^{125} - 1) + 1.$$

We have two inequalities for any $n, m \geq 1$: $\ell(n+1) \leq \ell(n) + 1$ and $\ell(nm) \leq \ell(n) + \ell(m)$. So, we can conclude that $\ell(2^{127} - 3) \leq \ell(2^{125} - 1) + 3$, since $\ell(4) = 2$.

An addition chain $a_0 = 1, a_1, \dots, a_{r-1}, a_r = n$ is called a “star chain” for n if for each $1 \leq i \leq r$ there exists $0 \leq j < i$ such that $a_i = a_{i-1} + a_j$. Denote by $\ell^*(n)$ the length of the shortest star chain for the number n . It is easy to see $\ell(n) \leq \ell^*(n)$. The following inequality holds by the famous Brauer’s theorem (Brauer 1939): $\ell(2^m - 1) \leq m - 1 + \ell^*(m)$ for any $m \geq 1$. The chain 1, 2, 3, 5, 10, 20, 25, 50, 100, 125 is a star chain for 125 of length 9. It is known [A003313] that $\ell(125) = 9$. So, the chain for 125 found above is the shortest one. Thus,

$$\ell(2^{127} - 3) \leq 125 - 1 + 9 + 3 = 136.$$

A required chain of length 136 for $2^{127} - 3$ is presented in Table 3.

There also exist several lower bounds that participants referred to.

Table 3. An addition chain of length 136 for $2^{127}-3$.

1, 2, 4, 8, 12, 24, 48, 60, 120, 124, 248, 496, 992, 1984, 3968, 4092, 8184, 16368, 32736, 65472, 130944, 261888, 523776, 1047552, 2095104, 4190208, 4194300, 8388600, 16777200, 33554400, 67108800, 134217600, 134217724, 268435448, 536870896, 1073741792, 2147483584, 4294967168, 8589934336, 17179868672, 34359737344, 68719474688, 137438949376, 274877898752, 549755797504, 1099511595008, 2199023190016, 4398046380032, 8796092760064, 17592185520128, 35184371040256, 70368742080512, 140737484161024, 281474968322048, 562949936644096, 1125899873288192, 2251799746576384, 4503599493152768, 4503599627370492, 9007199254740984, 18014398509481968, 36028797018963936, 72057594037927872, 144115188075855744, 288230376151711488, 576460752303422976, 1152921504606845952, 2305843009213691904, 4611686018427383808, 9223372036854767616, 18446744073709535232, 36893488147419070464, 73786976294838140928, 147573952589676281856, 295147905179352563712, 590295810358705127424, 1180591620717410254848, 2361183241434820509696, 4722366482869641019392, 9444732965739282038784, 18889465931478564077568, 37778931862957128155136, 75557863725914256310272, 151115727451828512620544, 302231454903657025241088, 604462909807314050482176, 1208925819614628100964352, 2417851639229256201928704, 4835703278458512403857408, 9671406556917024807714816, 19342813113834049615429632, 38685626227668099230859264, 77371252455336198461718528, 154742504910672396923437056, 309485009821344793846874112, 618970019642689587693748224, 1237940039285379175387496448, 2475880078570758350774992896, 4951760157141516701549985792, 9903520314283033403099971584, 1980704062856066806199943168, 39614081257132133612399886336, 79228162514264267224799772672, 158456325028528534449599515344, 31691265005705706889919990688, 633825300114114137798398181376, 126765060022828275596796362752, 2535301200456456551193592725504, 5070602400912913102387185451008, 5070602400912917605986812821500, 10141204801825835211973625643000, 20282409603651670423947251286000, 40564819207303340847894502572000, 81129638414606681695789005144000, 162259276829213363391578010288000, 32451855658426726783156020576000, 649037107316853453566312041152000, 1298074214633706907132624082304000, 2596148429267413814265248164608000, 5192296858534827628530496329216000, 10384593717069655257060992658432000, 20769187434139310514121985316864000, 41538374868278621028243970633728000, 83076749736557242056487941267456000, 166153499473114484112975882534912000, 332306998946228968225951765069824000, 664613997892457936451903530139648000, 13292279957849158729903807060279206000, 2658455991569831745807614120558592000, 5316911983139663491615228241117184000, 10633823966279326983230456482234368000, 21267647932558653966460912964468736000, 42535295865117307932921825928937472000, 85070591730234615865843651857874944000, 170141183460469231731687303715749888000, 170141183460469231731687303715884105724, 170141183460469231731687303715884105725.

1. First, one could notice that addition chains of length less than 127 cannot produce numbers greater than 2^{126} . So, we have $\ell(2^{127}-3) > 126$.
2. A more strict bound $\ell(2^{127}-3) > 132$ comes from Schönage's theorem (Schönage 1975):

$$\ell(n) \log_2(n) - \log_2(s(n)) - 2.13,$$

where $s(n)$ denotes the sum of the digits in the binary expansion of n .

3. Also, there is the famous Scholz-Brauer conjecture (Scholz 1937): $\ell(2^n-1) \leq n-1 + \ell(n)$ for any $n \geq 1$. Moreover, for all $n \leq 64$ the inequality becomes the equality as shown in Clift (2011). If we suppose that the conjecture is true and the equality always holds, then it can be assumed that $\ell(2^{127}-1) = 127-1 + 10 = 136$ since $\ell(127) = 10$ [A003313]. Then it is easy to see that $\ell(2^{127}-1) \leq \ell(2^{127}-3) + 1$. Thus, we have $\ell(2^{127}-3) \geq 135$, which is quite close to the shortest found length of 136.

At the end, thirteen teams in the second round were able to find addition chains of length 136 using different approaches and eight team presented chains of lengths 137 and 138.

Problem: “Hash function FNV2”

Formulation

The FNV2 hash function is derived from the function FNV-1a (FNV hash history 2013). FNV2 processes a message x composed of bytes $x_1, x_2, \dots, x_n \in \{0, 1, \dots, 255\}$ in the following way:

1. $h \leftarrow h_0$;

2. for $i = 1, 2, \dots, n$: $h \leftarrow (h + x_i)g \bmod 2^{128}$;
3. return h .

Here, $h_0 = 144066263297769815596495629667062367629$ and $g = 2^{88} + 315$.

Find a collision: that is, two different messages x and x' such that $\text{FNV2}(x) = \text{FNV2}(x')$. Collisions on short messages and collisions that are obtained without intensive calculations are welcomed. Supply your answer as a pair of two hexadecimal strings that encode bytes of colliding messages.

Solution

We provide a solution based on the Lenstra-Lenstra-Lovász (LLL) algorithm. This idea was proposed by several teams.

First, it is clear that

$$\text{FNV2}(x_1 x_2 \dots x_n) = (h_0 g^n + x_1 g^n + x_2 g^{n-1} + \dots + x_n g) \bmod 2^{128}.$$

Next, it is sufficient to solve the equation

$$z_1 g^{n-1} + z_2 g^{n-2} + \dots + z_n g^0 \equiv 0 \pmod{2^{128}}$$

in $z_1, z_2, \dots, z_n \in \{-255, \dots, 255\}$ not equal to zero simultaneously. Indeed, $z_i = x_i - y_i$ for some $x_i, y_i \in \{0, \dots, 255\}$ and

$$\begin{aligned} & \text{FNV2}(x_1, x_2, \dots, x_n) - \text{FNV2}(y_1, y_2, \dots, y_n) \\ &= g(z_1 g^{n-1} + z_2 g^{n-2} + \dots + z_n g^0) \equiv 0 \pmod{2^{128}}. \end{aligned}$$

The purpose is to construct a polynomial such that g is its root. Let us define integer vectors e^0, \dots, e^n of length $n+1$ in the following way:

$$e^0 = (\underbrace{0, \dots, 0}_n, t \cdot 2^{128}), \text{ where } t \text{ is a small integer,}$$

$$e^i = (\underbrace{0, \dots, 0}_{i-1}, 1, \underbrace{0, \dots, 0}_{n-i}, g^{n-i} \bmod 2^{128}), \text{ where } i \in \{1, \dots, n\}.$$

Let us add some z_0 to z_1, \dots, z_n and consider the linear combination

$$\ell_z = z_0 e^0 + \dots + z_n e^n = (z_1, \dots, z_n, z_0 t 2^{128} + z_1 g^{n-1} + z_2 g^{n-2} + \dots + z_n g^0).$$

To solve the problem it is sufficient to find a linear combination ℓ_z with $z_1, \dots, z_n \in \{-255, \dots, 255\}$ and $z_0 = 0$. This can be done using the LLL algorithm. It is a lattice-reduction algorithm that can find a short, nearly orthogonal basis of $\langle e^0, \dots, e^n \rangle$. Obtaining such an LLL-reduced basis, we check if it contains a vector ℓ_z with desired properties. According to the participants' results, this approach works well for some chosen (large) t and $n \approx 17$.

Message 1	Message 2
8080808080808080808080808080808080	a55eca84915f926b4a5f8146c78d8a75d893
8080808080808080808080808080808080	c07b375db56d8aceac504381d06696389f
8c2565bf035411600c3c0e20e21235	cb6163cf5f3
"~~~~ NSU CRYPTO IS FUN! ~~~~~"	82857b83274c57531e44524e49564b175351273f48572a1c79807c7a

Problem: “The image set” (unsolved)

Let \mathbb{F}_2 be a finite field with two elements and n be any positive integer. Let $g(X)$ be an irreducible polynomial of degree n over \mathbb{F}_2 . It is widely known that the set of equivalence classes of polynomials over \mathbb{F}_2 modulo $g(X)$ is a finite field of order 2^n ; we denote it by \mathbb{F}_{2^n} .

$$F(x) = x^3 + x.$$

An example: For $n=3$ we can take $g(X) = X^3 + X + 1$, then each element of the field \mathbb{F}_{2^3} can be written as a polynomial of degree at most two: $a_0 + a_1X + a_2X^2$, with $a_0, a_1, a_2 \in \mathbb{F}_2$. We can calculate the table of multiplication in \mathbb{F}_{2^3} modulo $g(X)$, while the table of addition just corresponds to adding polynomials over \mathbb{F}_2 . For example,

$$\begin{aligned} (1 + X + X^2) + (X + X^2) &= 1, \\ (X + X^2)(1 + X^2) &= X + X^2 + X^3 + X^4 = 1 + X \pmod{g(X)}. \end{aligned}$$

$$\{F(x)|x \in \mathbb{F}_{2^3}\} = \{0, 1, 1 + X, 1 + X^2, 1 + X + X^2\}.$$

You need to study this problem for an arbitrary n (or some partial cases).

Remarks: Functions over the finite field of order 2^n are of great interest for use in cryptographic applications; for example, as S-boxes. For instance, an AES S-box is based on the inverse function over \mathbb{F}_{2^8} . But in fact, there are

many open problems in fields of finding new constructions and descriptions of cryptographically significant functions!

Solution

There were no complete solutions for this problem. Some participants proposed nice ideas. Unfortunately, no one could push these ideas far enough to get significant results. Some did not understand what we were looking for (they focused on the number of solutions, which is known from Mullen and Panario 2013).

The best solution attempts were proposed by Alexey Udovenko (University of Luxembourg) and by the team of Nikolay Altukhov, Roman Chistiakov, and Evgeniy Manaka (Bauman Moscow State Technical University). The first solution characterized the case of one preimage (that is classical), showed a property by the algebraic degree (which gives weak insight on the structure, but it was a nice idea), and finished with observations that are nice but not specific. The second one had an idea of using the greatest common divisor and tried to calculate it, but did not complete it.

We would also like to recall a known result that may be useful for solving the problem.

Theorem. Let t_1, t_2 denote the roots of $t^2 + bt + a^3 = 0$ in $\mathbb{F}_{2^{2n}}$, where $a \in \mathbb{F}_{2^n}, b \in \mathbb{F}_{2^n}^*$ (Williams 1975). Then the factorization of $f(x) = x^3 + ax + b$ over \mathbb{F}_{2^n} is characterized as follows:

- f has three zeros in \mathbb{F}_{2^n} if and only if $tr_n(\frac{a^3}{b^2} + 1) = 0$, where tr_n is the absolute trace function and t_1, t_2 are cubes in \mathbb{F}_{2^n} (n even), $\mathbb{F}_{2^{2n}}$ (n odd).
- f has exactly one zero in \mathbb{F}_{2^n} if and only if $tr_n(\frac{a^3}{b^2} + 1) = 1$.
- f has no zero in \mathbb{F}_{2^n} if and only if $tr_n(\frac{a^3}{b^2} + 1) = 0$ and t_1, t_2 are not cubes in \mathbb{F}_{2^n} (n even), $\mathbb{F}_{2^{2n}}$ (n odd).

This result depends on t_1 and t_2 and, when $b \neq 0$, the change of variable $x = bt$ transforms the equation $t^2 + bt + a^3 = 0$ into the equation $x^2 + x = \frac{a^3}{b^2}$. So, it may be useful to recall the following fact.

Theorem. Let n be any positive integer and $\beta \in \mathbb{F}_{2^n}$ (Zinoviev 1996). A necessary and sufficient condition for the existence of solutions in \mathbb{F}_{2^n} of the equation $x^2 + x = \beta$ is that $tr_n(\beta) = 0$. Assuming that this condition is satisfied, the solutions of the equation are $x = \sum_{j=1}^{n-1} \beta^{2^j} (\sum_{k=0}^{j-1} c^{2^k})$ and $x = 1 + \sum_{j=1}^{n-1} \beta^{2^j} (\sum_{k=0}^{j-1} c^{2^k})$, where c is any (fixed) element such that $tr_n(c) = 1$.

Problem: “Boolean hidden shift and quantum computings” (unsolved)

Formulation

The following longstanding problem is known. Let $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$ be a given Boolean function. Determine the hidden nonzero shift $a \in \mathbb{F}_2^n$ for the function (i.e., a vector such that $f_a(x) = f(x \oplus a)$ for all $x \in \mathbb{F}_2^n$). This should be done while it has limited access to an oracle for the shifted Boolean function f_a with unknown shift a (i.e., a black box, which computes the function $f(a \oplus x)$ for a given vector x). Such a problem is called the Boolean hidden shift problem (BHSP).

To solve this problem on a quantum computer, an oracle that computes the shifted function in the phase is used. This oracle can be implemented using only one query to an oracle that computes the function in a register. The phase oracle is a unitary operator defined by its action on the computational basis $O_{f_a} : |x\rangle \mapsto (-1)^{f(x \oplus a)} |x\rangle$, where $|x\rangle$ is the index register. The quantum query complexity is the minimum number of oracle O_{f_a} accesses needed in the worst case to solve the problem.

There are two classes of Boolean functions for which the quantum query complexity is minimal and maximal, respectively:

- for any bent function (i.e., a function with an even number of variables that is on the maximal possible Hamming distance from the set of all affine functions) one quantum query suffices to solve the problem exactly (Roetteler 2010);
- for any delta function (i.e., $f(x) = \delta_{x,x_0}$ for some $x_0 \in \mathbb{F}_2^n$) the quantum query complexity is $\Theta(2^{n/2})$, which is equivalent to Grover’s search (Bennett et al. 1997; Grover 1996).

For any Boolean function f in n variables $Q(\text{BHSP}_f) = O(2^{n/2})$, where $Q(\text{BHSP}_f)$ is the bounded error quantum query complexity of the BHSP for f . Moreover, it holds (Childs et al. 2013)

$$Q(\text{BHSP}_f) \leq \frac{\pi}{4} \frac{2^{n/2}}{\sqrt{\text{wt}(f)}} + O\left(\sqrt{\text{wt}(f)}\right),$$

when $1 \leq \text{wt}(f) \leq 2^{n-1}$ and where $\text{wt}(f)$ is the Hamming weight of f .

The problem to solve is the following: Identify natural classes of Boolean functions in an even number of variables lying between the two extreme cases of bent and delta functions and characterize the quantum query complexity of the BHSP for these functions (Childs et al. 2013).

Solution

The Boolean hidden shift problem is a particular noninjective case of the well-known hidden shift problem. There were no complete solutions for this problem. Some attempts to use known results from quantum computation, including quantum certificate complexity, were made by the team of Andrey Kalachev, Danil Cherepanov, and Alexey Radaev (Bauman Moscow State Technical University), but no detailed descriptions of classes of Boolean functions with query complexity of BHSP distinct from two known extremal cases were given.

Problem: “Useful proof of work for blockchains” (unsolved)

Formulation

A proof-of-work system is one of the key parts of modern blockchain-based platform implementations, such as the cryptocurrencies **Bitcoin** or **Ethereum**. Proof-of-work means that the user is required to perform some work in order to request some service from the system (e.g., to send an email or create a new block of transactions for the blockchain).

For example, in the Bitcoin system, if some user wants to create a block of transactions and add it to the chain, the hash value of his block must satisfy certain conditions that can be achieved by iterating a special variable X inside the block many times and checking the resulting hash value on every iteration.

What is important about the problem in a proof-of-work system is the following:

- It is known that the solution for the problem exists, and it is also known how many iterations (on average) are required to find it using the best-known algorithm \mathcal{A} ;
- There are no algorithms for solving the problem that perform significantly better than \mathcal{A} ; it is also believed that such algorithms will not soon be found;
- The problem depends on some input data I , so you cannot find solutions for the problem in advance (before input I is known) and then use these solutions without performing any work;
- Given a problem and a solution to it, it is easy to verify that the provided solution is correct.

Unfortunately, solving the problem of finding specific hash values (used in Bitcoin and Ethereum) does not yield any information that is useful outside the system; therefore, tremendous amounts of calculations performed to solve the problem are wasted.

Some other implementations of proof-of-work systems solve this issue. For example, solutions for proof-of-work problems used in the cryptocurrency **Primecoin** give us special chains of prime numbers that are useful for scientific research.

Your task is to construct a problem **P** that can be used in a proof-of-work system such that information obtained in the process of solving it can be useful outside the system. More formally:

- **P** is, in fact, a family of problems parametrized by two variables: I (input data, you can assume that I is a 256 bit string, or introduce other sensible formats), and C (complexity, e.g., some positive integer). For fixed input and complexity, $\mathbf{P}(I, C)$ is a problem that can be solved by using some algorithm \mathcal{A} (should be provided in your solution to this task). It should not be possible to find a provable solution for the problem $\mathbf{P}(I, C)$ if I is not known;
- Average time T (amount of computational steps or iterations) required to find a solution of $\mathbf{P}(I, C)$ using algorithm \mathcal{A} is known (assuming input data I is chosen randomly and uniformly) and depends on C , so $T = T(C)$ and $T(C)$ can be made very small, infeasibly large, or something in between by adjusting complexity variable C ;
- It should be easy to verify whether any provided solution is correct or not;
- Any kind of proof that there are likely no significantly better algorithms for solving **P** than the given algorithm \mathcal{A} is desirable. For example, a proof that proposed the problem is NP-hard or any other considerations;
- You should describe how information obtained in the process of solving **P** can be useful outside of the proof-of-work system.

For example, in the **Bitcoin** system $\mathbf{P}(I, C)$ is a problem of finding an integer X such that if we apply the SHA-256 hash function to the pair (I, X) twice, the resulting hash value, represented as an integer, will not be greater than C . Here C is a nonnegative integer that defines the complexity of the problem and I —a block header containing information about all transactions included in it along with some other information—is an input.

Solution

There were no complete solutions for this problem. Many contestants proposed using NP-hard problems for proof-of-work, but no detailed descriptions of how to convert a hard problem into a proof-of-work were provided. In some solutions input data were not linked with the problem. In other solutions the condition of “easy verifiability” of the solution to the

Table 5. Winners of the first round in school section A (in the category “school student”).

Place	Name	Country, City	School	Score
1	Alexander Grebennikov	Russia, Saint Petersburg	Presidential PML 239	22
1	Ivan Baksheev	Russia, Novosibirsk	Gymnasium 6	21
2	Alexander Dorokhin	Russia, Saint Petersburg	Presidential PML 239	18
3	Vladimir Schavelev	Russia, Novosibirsk	SESC NSU	17
3	Borislav Kirilov	Bulgaria, Sofia	FPMG	17
Diploma	Ana Kapros	Romania, Rm Valcea	National College Mircea cel Batran	10
Diploma	Filip Dashtevski	Macedonia, Kumanovo	Yahya Kemal College	10
Diploma	Andrei Razvan	Romania, Craiova	“Fratii Buzesti” National College	10
Diploma	Stefan Zaharia	Romania, Vaslui	Lyceum Mihail Kogalniceanu	9
Diploma	Ilia Krytsin	Russia, Novosibirsk	SESC NSU	9
Diploma	Grigorii Popov	Russia, Novosibirsk	SESC NSU	8
Diploma	Bogdan Circeanu	Romania, Craiova	“Fratii Buzesti” National College	8
Diploma	Lenart Bucar	Slovenia, Grosuplje	Gymnasium Bezigrad	7
Diploma	Maxim Desyatkov	Russia, Kuibyshev	SESC NSU	7

Table 6. Winners of the first round, section B (in the category “university student”).

Place	Name	Country, City	University	Score
1	Roman Lebedev	Russia, Novosibirsk	Novosibirsk State University	20
2	Robert Spencer	United Kingdom, Cambridge	University of Cambridge	15
3	Nikita Odinkih	Russia, Novosibirsk	Novosibirsk State University	14
3	Alexey Miloserdov	Russia, Novosibirsk	Novosibirsk State University	14
3	Dheeraj M. Pai	India, Chennai	Indian Institute of Technology, Madras	13
Diploma	Alexey Solovov	Russia, Moscow	Lomonosov Moscow State University	10
Diploma	Khai Hanh Tang	Vietnam, Ho Chi Minh	University of Science	9
Diploma	Evgeniy Manaka	Russia, Moscow	Bauman Moscow State Technical University	9
Diploma	Andrey Klyuev	Russia, Moscow	National Research Nuclear University MEPhI	8
Diploma	Nikolay Altukhov	Russia, Moscow	Bauman Moscow State Technical University	8
Diploma	Vladimir Bushuev	Russia, Korolev	Bauman Moscow State Technical University	8
Diploma	Roman Chistiakov	Russia, Moscow	Bauman Moscow State Technical University	8
Diploma	Mikhail Sorokin	Russia, Moscow	National Research Nuclear University MEPhI	8

problem was not satisfied: it is not easy to check for a “no” answer in an NP-hard decision problem.

An interesting approach for constructing useful proof-of-work was proposed by Carl L ndahl (Sweden). Suppose that we ask a user to solve two problems, say P_1 and P_2 . Problem P_1 is a regular hash-seeking problem such as the one used in **Bitcoin**. It is easy to link an input with such a problem in order to adjust its difficulty. Problem P_2 is some problem based on an NP-hard problem, and P_2 input is based on the solution of P_1 . P_2 that can have larger variance in time complexity, but we can make average time complexity be dominated by P_1 , thus keeping overall proof-of-work time consumption more consistent. At the same time, we will be obtaining solutions for some NP-hard problems in the process of obtaining proof of work.

Winners of the Olympiad

Please see [Tables 5–10](#) for the names of and information about the winners of NSUCRYPTO-2017.

Table 7. Winners of the first round, section B (in the category “professional”).

Place	Name	Country, City	Organization	Score
1	Alexey Udovenko	Luxembourg, Luxembourg	University of Luxembourg	28
2	Henning Seidler	Germany, Berlin	TU Berlin	16
2	George Beloshapko	Switzerland, Zürich	Google	15
3	Daniel Malinowski	Poland, Warsaw	University of Warsaw	12
Diploma	Evgeniya Ishchukova	Russia, Taganrog	Southern Federal University	8
Diploma	Egor Kulikov	Germany, Munich	dxFeed Solutions GmbH	8

Table 8. Winners of the second round (in the category “School Student”).

Place	Name	Country, City	School	Score
Diploma	Filip Dashtevski, Gorazd Dimitrov	Macedonia, Kumanovo	Yahya Kemal College	8
Diploma	Amalia Rebegea, Gabi Tulba-Lecu, Stefan Manolache	Romania, Bucharest	CNI “Tudor Vianu”	6

Table 9. Winners of the second round (in the category “university student”).

Place	Name	Country, City	University	Score
1	Roman Lebedev, Vladimir Sitnov, Ilia Koriakin	Russia, Novosibirsk	Novosibirsk State University	50
2	Alexey Miloserdov, Nikita Odinokih, Saveliy Skresanov	Russia, Novosibirsk	Novosibirsk State University	46
2	Maxim Plushkin, Ivan Lozinskiy, Azamat Miftakhov	Russia, Moscow	Lomonosov Moscow State University	44
3	Irina Slonkina	Russia, Moscow	National Research Nuclear University MEPhI	38
3	Ngoc Ky Nguyen, Thanh Nguyen Van, Phuoc Nguyen Ho Minh	Vietnam, Ho Chi Minh City	Bach Khoa University, Ho Chi Minh University of Technology	34
3	Nikolay Altukhov, Roman Chistiakov, Evgeniy Manaka	Russia, Moscow	Bauman Moscow State Technical University	32
Diploma	Mikhail Sorokin, Andrey Klyuev, Anatoli Makeyev	Russia, Moscow	National Research Nuclear University MEPhI	26
Diploma	Oskar Soop, Joosep Jääger, Andres Unt	Estonia, Tartu	University of Tartu	26
Diploma	Andrey Kalachev, Danil Cherepanov, Alexey Radaev	Russia, Moscow	Bauman Moscow State Technical University	24
Diploma	Dianthe Bose	India, Chennai	Chennai Mathematical Institute	23
Diploma	Mikhail Kotov, Oleg Zakharov, Sergey Batunin	Russia, Tomsk	Tomsk State University	20

Table 10. Winners of the second round (in the category “professional”).

Place	Name	Country, City	Organization	Score
1	Alexey Udovenko	Luxembourg, Luxembourg	SnT, University of Luxembourg	63
2	Daniel Malinowski, Michal Kowalczyk	Poland, Warsaw	University of Warsaw, Dragon Sector	49
3	Alexey Ripinen, Oleg Smirnov, Peter Razumovsky	Russia, Saratov	Saratov State University	40
3	Duc Tri Nguyen, Dat Bui Minh Tien, Quan Doan	Vietnam, Ho Chi Minh city	CERG at George Mason University, Meepwn CTF Team, MeePwn	37
3	Carl Londahl	Sweden, Karlskrona	Independent researcher	31
Diploma	Anna Taranenko	Russia, Novosibirsk	Sobolev Institute of Mathematics	29
Diploma	Kristina Geut, Sergey Titov	Russia, Yekaterinburg	Ural State University of Railway Transport	28
Diploma	Victoria Vlasova, Mikhail Polyakov, Mikhail Tsvetkov	Russia, Moscow	Bauman Moscow State Technical University	26
Diploma	Harry Lee, Samuel Tang	Hong Kong, Hong Kong	Hong Kong University of Science and Technology	20
Diploma	Henning Seidler, Katja Stumpp	Germany, Berlin	Berlin Technical University	20

Funding

The paper was supported by the Russian Ministry of Science and Education (the 5-100 Excellence Programme and Project No. 1.12875.2018/12.1), by the Russian Foundation for Basic Research (Project Nos. 17-41-543364, 18-07-01394, 18-31-00479, and 18-31-00374), and by the Program of Fundamental Scientific Research of the SB RAS no.I.5.1. (Project No. 0314-2016-0017).

ORCID

A. Gorodilova  <http://orcid.org/0000-0002-4986-434X>

References

- Agievich, S., A. Gorodilova, V. Idrisova, N. Kolomeec, G. Shushuev, and N. Tokareva. 2017. Mathematical problems of the second international student’s olympiad in cryptography. *Cryptologia* 41 (6):534–65.
- Agievich, S., A. Gorodilova, N. Kolomeec, S. Nikova, B. Preneel, V. Rijmen, G. Shushuev, N. Tokareva, and V. Vitkup. 2015. Problems, solutions and experience of the first international student’s olympiad in cryptography. *Prikladnaya Diskretnaya Matematika (Applied Discrete Mathematics)* 10 (3):41–62.
- Bennett, C. H., E. Bernstein, G. Brassard, and U. Vazirani. 1997. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing* 26 (5):1510–23.

- Bhasin, S., C. Carlet, and S. Guilley. 2013. Theory of masking with codewords in hardware: low-weight dth-order correlation-immune boolean functions. *IACR Cryptology ePrint Archive, Report* 303:1–37.
- Bierbrauer, J. 1995. Bounds on orthogonal arrays and resilient functions. *Journal of Combinatorial Designs* 3 (3):179–83.
- Boneh, D. 1999. Twenty years of attacks on the RSA cryptosystem. *Notices AMS* 17 (2): 203–13.
- Brauer, A. 1939. On addition chains//. *Bulletin of the American Mathematical Society* 45 (10):736–9.
- Canteaut, A. 2016. Lecture Notes on Cryptographic Boolean Functions. <https://www.rocq.inria.fr/secret/Anne.Canteaut/poly.pdf>.
- Childs, A. M., R. Kothari, M. Ozols, and M. Roetteler. 2013. Easy and hard functions for the boolean hidden shift problem. *Proceedings of TQC 2013* 22:50–79.
- Clift, N. M. 2011. Calculating optimal addition chains. *Computing* 91 (3):265–84.
- Daemen, J., and V. Rijmen. 2002. *The design of rijndael: AES — the advanced encryption standard*. Berlin, Germany: Springer, 238.
- FNV hash history. 2013. <http://www.isthe.com/chongo/tech/comp/fnv/>
- Friedman, J. 1992. On the bit extraction problem//Proc. 33rd IEEE Symposium on Foundations of Computer Science. P. 314–319.
- Geut, K., K. Kirienko, P. Sadkov, R. Taskin, and S. Titov. 2017. On explicit constructions for solving the problem “a secret sharing”. *Prikladnaya Diskretnaya Matematika. Prilozhenie* 10:68–70.
- Grover, L. K. 1996. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC 1996)*. P. 212–219.
- Hastad, J. 1988. Solving simultaneous modular equations of low degree. *SIAM Journal on Computing* 17 (2):336–41.
- Hedayat, A. S., N. J. A. Sloane, and J. Stufken. 1999. *Orthogonal arrays, theory and applications*. *Springer series in statistics*. New York, NY: Springer.
- MacWilliams, F. J., and N. J. A. Sloane. 1977. *The theory of error-correcting codes*. New York, NY: North-Holland Publishing Company
- Mullen, G. L., and D. Panario. 2013. *Handbook of finite fields*. Boca Raton, FL: Chapman & Hall/CRC.
- Picek, S., S. Guilley, C. Carlet, D. Jakobovic, and J. F. Miller. 2015. Evolutionary approach for finding correlation immune boolean functions of order t with minimal hamming weight. In: Dediu AH., Magdalena L., Martn-Vide C. (eds) *Theory and practice of natural computing. Lecture notes in computer science*. 9477. New York, NY: Springer.
- Rivest, R. L., A. Shamir, and L. Adleman. 1978. A method for obtaining digital signatures and public key cryptosystems. *Communications of the ACM* 21 (2):120–6.
- Roetteler, M. 2010. Quantum algorithms for highly non-linear Boolean functions. *Proceedings of the 21st ACM-SIAM Symposium on Discrete Algorithms (SODA 2010)*. P. 448–457.
- Schönhage, A. 1975. A lower bound for the length of addition chains. *Theoretical Computer Science* 1 (1):1–12.
- Scholz, A. 1937. Jahresbericht. *Deutsche Mathematiker Vereinigung* 252 (47):41–2.
- Sloane, N. J. A. 1964. A003313 — Length of shortest addition chain for n //The On-Line Encyclopedia of Integer Sequences. <https://oeis.org/A003313>.
- Tokareva, N., A. Gorodilova, S. Agievich, V. Idrisova, N. Kolomeec, A. Kutsenko, A. Oblaukhov, and G. Shushuev. 2018. Mathematical methods in solutions of the problems

- from the third international students' olympiad in cryptography. *Prikladnaya Diskretnaya Matematika (Applied Discrete Mathematics)* 40:34–58.
- Wei, T. J. 2018. 2017 is not just another prime number. <http://weijr-note.blogspot.ru/2017/01/2017-is-not-just-another-prime-number.html>
- Williams, K. 1975. Note on cubics over $GF(2^n)$ and $GF(3^n)$. *Journal of Number Theory* 7 (4):361–5.
- Zinoviev, V. 1996. On the Solution of Equations of Degree. [Research Report] RR-2829, INRIA. (inria-00073862)