The Condenser PBJ cipher machine

Eugen Antal, Paul Reuvers, Pavol Zajac

2024

This is an Accepted Manuscript of an article published by Taylor & Francis in Cryptologia, available at: https://doi.org/10.1080/01611194.2024.2372254

Abstract

Condenser PBJ is a mechanical cipher machine developed by Pavel Baráček-Jacquier (1885-1969). It was created between 1922 and 1924, and was used by the Czechoslovak diplomatic service until 1934. The machine was designed to be used as a tool for condensing telegraphic messages in a similar way to a codebook, but with encryption added to the result. The encryption algorithm is a polyalphabetic substitution cipher — an autokey cipher with a priming key of length 10. This autokey cipher is different however from the classical Vigenère autokey cipher. We have been able to fully reconstruct the encryption algorithm. We also present a detailed description of the machine, and point out some weaknesses in the cipher's design.

1 Introduction

The wide adaptation of various cipher machines can be dated to the 1920s. Most of these machines mechanized a polyalphabetic substitution method in a practical way. Czechoslovakia was also involved in the development of this type of cryptographic machine. The first cipher machine used in the Czechoslovak Army was Štolba (Antal and Zajac, 2023). It was developed by Josef Sieber (Štolba) in the mid-1930s and was issued to the Czechoslovak Army in 1938. During World War II (WW2) it was used in Slovakia by the army and by the diplomatic services. We originally thought that Štolba was the first Czechoslovak cipher machine used in practice (not only in the army), but we were mistaken. Some documents obtained from archives (VUA, n.d.; AMZV, n.d.) mention an earlier encryption machine developed

in Czechoslovakia, but for a long time, we were unable to get any further information about this machine. Recently we have found documents from the archives about a cipher machine developed in Czechoslovakia in the 1920s called "Condenser PBJ".

Condenser PBJ is a mechanical cipher machine developed by Pavel Baráček-Jacquier[1] in 1922-1924 (AMZV, 1934), which was used in Czechoslovak diplomacy (Ministry of Foreign Affairs and embassies) from 1924 onwards. The name 'Condenser' is a bit misleading as we will see later, whilst 'PBJ' is formed from the inventor's initials. Based on (VUA, n.d.), before the proclamation of Czechoslovakia in 1918, Baráček worked in Geneva, where he started to develop his first cipher machine (around 1915), which was later perfected and subsequently used by the Ministry of Foreign Affairs in Czechoslovakia[2]. The Condenser PBJ cipher machine was used by the Czechoslovak diplomatic service until the end of the year 1933.

On 1 January 1934, the new Telegraphic Regulations came into effect, as a result of which the machine had to be abandoned (ITU, 1932). Although in his memorandum (AMZV, 1934) Baráček doesn't say why, there are two articles that might have been responsible for this. According to article 10 §2, code words should not contain more than five letters. This could have been circumvented however, by sending each of the PBJ's 10-letter code words as two 5-letter groups. Article 11 §2 is more restrictive, as it forbids the mixed use of letters and figures in secret language, which is something the PBJ machine allows.

In the following sections, we provide more details about the Condenser PBJ cipher machine. We start with a short introduction to telegraphic communication and code condensers. We continue with the detailed description of the Condenser PBJ machine's main mechanical components and operations, respectively. Despite the fact that the preserved documents do not contain all details of the machine, we were able to fully reconstruct the encryption algorithm and the cipher keys from preserved plaintext-ciphertext pairs.

[1]Pavel Baráček-Jacquier (1885-1969) worked for the Ministry of Foreign Affairs in Czechoslovakia from 1920. He was also responsible for developing the cipher department (section B) of the Ministry of Foreign Affairs in Czechoslovakia in the years 1921-1925.

[2]The cipher machine's name is not mentioned, but from the documents, it is highly probable that he was pointing to an early version of the Condenser PBJ cipher machine.

2 Code condensers and radio-telegraphy

In commercial radio telegraphy, code condensers were used for the compression of large numbers and frequently used phrases by sending them as 5-letter groups, as demonstrated in (Peterson, 1929). This reduced the number of characters and, hence, the cost of a telegram. In addition, it allowed the creation of pronounceable 5- and 10-letter code words, which were easier for the telegraph operators to handle. Note that in those days, telegrams had to be transferred either verbally, by means of morse code or via teletypewriter. Sending a message over a long distance, for example in international trade, required it to be retyped several times. As pronounceable words like VEHICLE are easier to memorize than arbitrary code words like PBQFZ, the operator handled them faster and with fewer mistakes. As a result, customers who used pronounceable codewords were given a discount. Furthermore, it allowed a certain level of error-detection and in some cases even error-correction (McVey, 2013). Good examples of code words with built-in redundancy for error-correction can be found in The New Boe Code of 1937 (Boe Code, 1937). Pronounceable words were not just restricted to real words, but also included invented words like OXIDONU and OBURATEM.

The PBJ machine was designed to be used as a code condenser or, more precisely, a tool to reduce telegraphic expenses[3]. For this reason, a special format was required: in the input text, consonants and vowels (a, e, i, o, u) had to be alternated. Consonants were allowed in odd positions only, whilst vowels were only allowed in even positions. Although this doesn't make the message shorter (on the contrary: it makes the message longer), it makes the code words pronounceable — in the documents this is called phonetized text — as a result of which they qualify for a discount and will therefore be cheaper to send. The text, which could also contain numbers, was divided into blocks of 10 characters (padded with null characters if necessary). The PBJ machine encrypts these 10-letter code words without affecting their format, as each consonant is substituted by a consonant and each vowel is replaced by a vowel. In the context of the following descriptions, a combination of one consonant and one vowel will be called a *syllable*. This means that each 10-letter code word consists of 5 syllables.

We briefly summarize the rules[4] based on the original Condenser PBJ operating instructions (NARA, 1924).

[3]Baráček in (AMZV, n.d.) stated that with the condenser cipher machine, the ministry was able to reduce the telegraphic expenses by 40%, based on (AMZV, 1934) it was 50%.

[4]We have omitted one rule from the manual which refers to another encryption directive that we do not have at our disposal.

- Rule 1: Numbers in the text must be formatted as follows (except for the first group):
  - Insert the word *NUME* before the number.
  - Insert the number of digits of the number.
  - Write the numeric value of the number.
  - Insert the checksum (sum of digits mod 10).
- Rule 2: Selected bigrams[5], letters, and special characters are replaced:
  - př, pr → f
  - št, st → q
  - ch → g
  - šk, sk, čk, ck, → w
  - y → i
  - the dot and comma characters → ko
  - question mark character → co
- Rule 3: Alternation of consonants and vowels. If two consonants are next to each other, it is necessary to add the vowel U between them. If two vowels are next to each other, it is necessary to add consonant X between them. Letters U and X are nulls in this context, however, using these letters is not omitted. For this reason, their nature as nulls must be deduced/guessed from the context during decryption. [6]
- Rule 4: If the text begins with a vowel, consonant X must be added as the first letter. If the text ends with a consonant, the vowel U must be inserted at the end.
- Rule 5: If the last group's length is less than 10 characters after application of Rule 4, additional characters must be added. First, repeat the last syllable of the group. If more letters are missing, insert any additional syllables (Rule 3 must be retained).

All diacritical characters are removed from the formatted plaintext.

[5]Frequent bigrams are replaced with less frequent letters.

[6]Because the text can also contain numbers, it is necessary to check if in the formatted text only consonants are in odd places, and only vowels in even places. If not, additional nulls (U and X) must be inserted.

3 Machine Description

The encryption algorithm is a polyalphabetic substitution cipher - autokey cipher with a priming key of length 10. The machine is used as an encryption aid[7] for the calculation. It performs parallel addition of the key to the plaintext on blocks of length 10. The autokey is derived from the ciphertext, where the first block of the ciphertext (initialization vector) is specially constructed (section 3.3). However, this autokey cipher is different from the classical Vigenère autokey cipher. For more details of the encryption algorithm see section 4.

The machine's main components (NARA, 1924) were:

- Body (containing 10 individual mechanical adding machines)
- Strips (10 removable paper strips, numbered 0-9)
- Selectors (10 letter selection knobs with 10 positions each)
- Rotors (10 rotors, each visible through a window)
- Reset lever
- Crank

Two of these components – the *strips* and the *rotors* – are secret. When the machine is not in use, the secret components have to be stored in a safe to prevent unauthorized access. Without the secret components, the machine is not considered a controlled cryptographic item (CCI) and can be handled by an unauthorized person, for example for repair.

[7]The encryption can be performed directly on paper without the need of the cipher machine. Only the knowledge of the machine settings (the strips) is required. According to (NARA, 1924) we can consider the machine as a calculating machine, which can be repaired in case of failure (after removal of secret components) directly by the repairers of calculating machines.

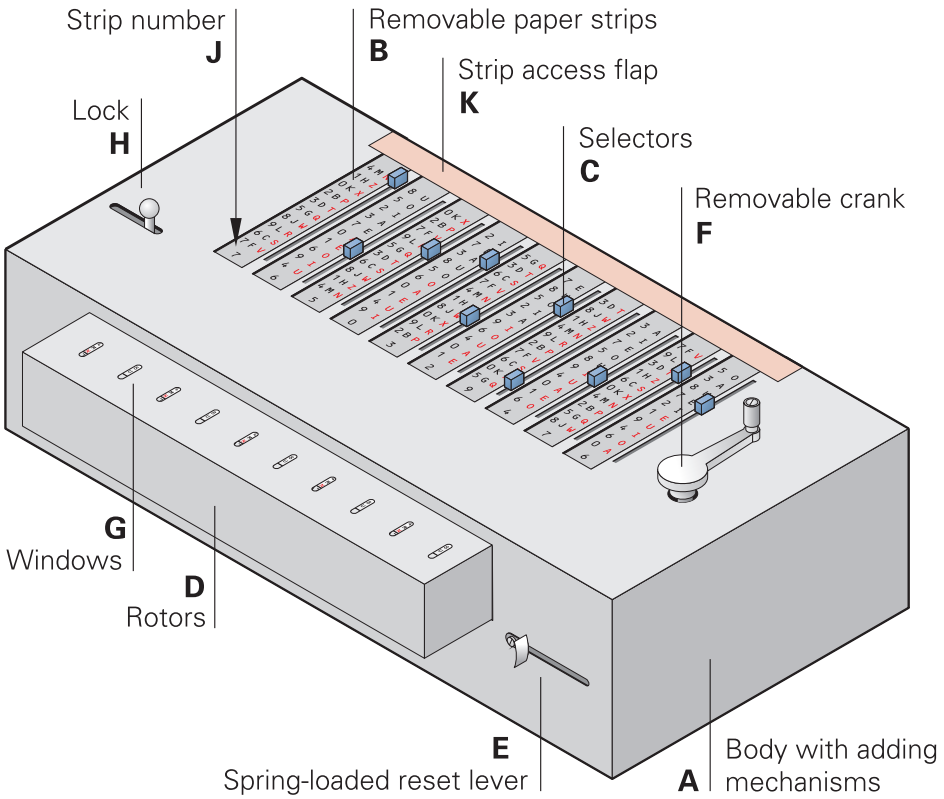


Figure 1: Educated guess of the machine's exterior

Figure 1, the body (A) holds the base of the adding machines, into which the secret components – the strips (B) and the rotors (D) – are inserted in a given order, indicated by the daily key. The strips are inserted successively into the slots between the selectors (C). The machine uses five vowel strips and five consonant strips. The daily key must be constructed in such a way that the consonant/vowel strips are alternated, starting with a consonant strip. This is necessary because only consonant strips can be used to encrypt consonants and vowel strips to encrypt vowels (see the text formatting rules in Section 2). The specific strips are determined by the annual key and are fixed for a long period of time, only their order is changed depending on the daily key (see Section 3.5 for more information).

At the front of the machine are ten rotors (D), of which the current positions are visible through ten rectangular windows (G) in the top of their protective cover. Each rotor has 10 faces that are printed with the same letters and figures as the corresponding strip.

The machine works on a block of 10 characters at once. The selectors (C) are used to set the input letters on the strips. The reset lever (E) is used to

clear the machine's inner state at the beginning of the encryption process, after the first group (the initialization vector, or IV) has been set with the selectors. By pushing the spring-loaded reset lever to the right, all rotors are set to their neutral position. The crank (F) is used to set a new inner state each time a new group of letters is entered by means of the selectors.

It should be stressed, that the machine itself does not perform any cryptographic operation. It merely acts as an aid to add the internal state of the machine to the current 10 input characters, and show the result of the addition on the rotors.

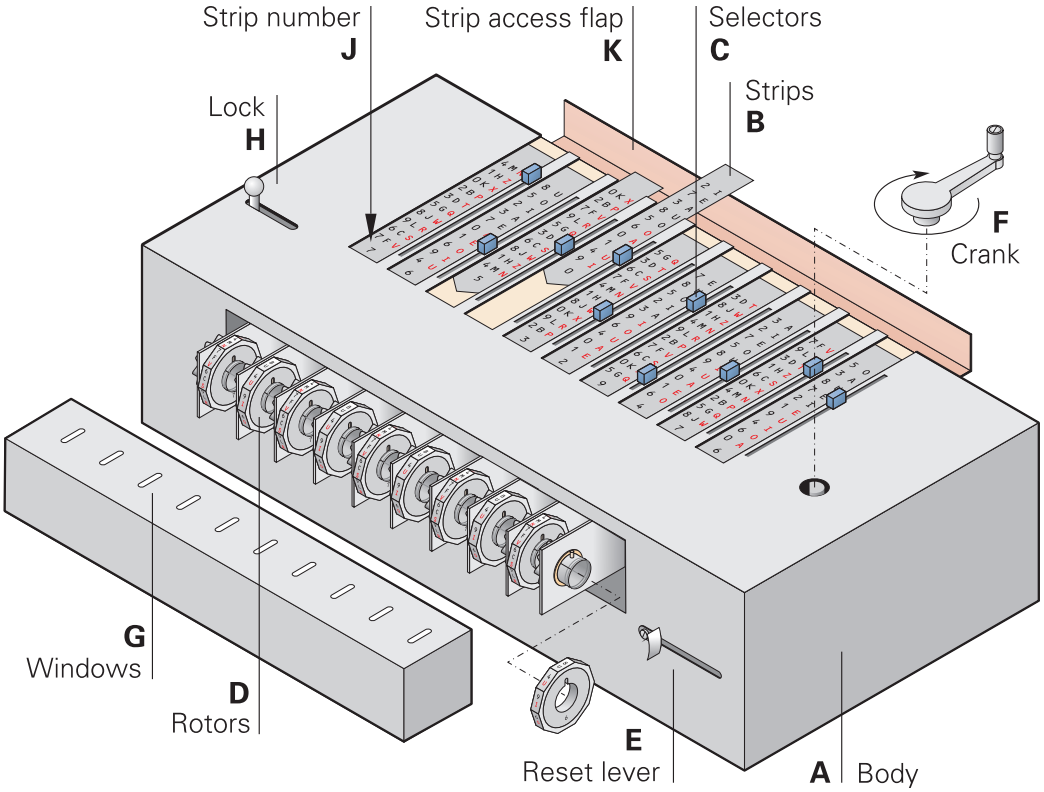


Figure 2: PBJ machine with strip access flap open and rotor cover removed

After unlocking the machine by pushing the lock (H) to the left, the strip access flap (K) can be tilted towards the rear, after which the strips (B) can be removed (Figure 2). Furthermore, the protective rotor cover at the front can be removed to reveal the ten rotors. Each rotor has to be installed directly under the corresponding strip (i.e. the strip with the same letters and figures).

Inside the body of the machine, are ten individual mechanical adding machines – one for each letter – comparable to a mechanical cash register

of the era, but without carry. Each time the crank is rotated, the value set with a selector (0-9) is added to the corresponding rotor by rotating it a number of steps equal to the value. Due to the circular nature of the rotor, it automatically performs a modulo 10 addition. A detailed description of a mechanical adder is beyond the scope of this article.

3.1 Encryption process

Before encryption, the cipher machine must be set up: strips are inserted based on the daily key, and the corresponding rotors are installed underneath them. Next, the crank is mounted at the top surface of the machine. It is used to advance the internal mechanism to the next state. Turning the crank *right*, means rotating it 360° clockwise. Turning it *left* means rotating the crank 360° counter clockwise.

The encryption process consists of the following steps:

1. Format the plaintext based on the formatting rules (Section 2).
2. Construct the first group of the ciphertext. This is the initialization vector (IV) (Section 3.3).
3. Set the letters of the first ciphertext group (IV) on the strips using the selectors. (The IV is sent *in clear*.)
4. Engage the reset lever, release it, and turn the crank right.
5. Set the letters of the first plaintext group on the strips using the selectors.
6. Turn the crank right.
7. Read the new ciphertext group from the windows at the front of the machine.
8. For the remaining groups repeat:
   (a) Set the letters of the next plaintext group on the strips using the selectors.
   (b) Turn the crank right.
   (c) Read the new ciphertext group from the windows.

The inner state of the machine is set by entering the letters (all ten letters of the selected group) using the selectors (C) and turning the crank right by 360°. This operation will add the ten letters (in parallel) to the previous state and create the new state. In the encryption process, only the first ciphertext group (IV) is set on the machine at the beginning, and the reset lever – followed by a right turn of the crank – is used to set this state as the initial state. Next, the first plaintext group is entered using the selectors, and then added to the first ciphertext group by turning the crank right. When a plaintext group is added to the ciphertext group, the resulting new ciphertext automatically becomes the inner state, so it is not required to be set. Therefore, only the plaintext groups are repeatedly entered using the selectors (C) in the successive steps.

3.2 Decryption process

The decryption procedure is almost the same as the encryption procedure (Section 3.1) and differs only in a few steps. Because the ciphertext is already formatted, step **1** is skipped. During the encryption process, the crank had to be turned right to add the input (plaintext) to the previous state. When decrypting however, the input (ciphertext) has to be subtracted from the previous state. Therefore, in steps **4** and **8(b)**, the crank has to be turned *left* instead of right. Note however that in step **6** the crank must still be turned right.

The encryption/decryption process can also be calculated directly on paper without the need for the machine. We provide a detailed description of this process in Appendix B. The machine is merely an aid to do it faster and with less chance of making mistakes.

3.3 Initialization vector

The first group of the ciphertext is the initialization vector (IV). This group is used in the encryption/decryption process as an initial inner state of the cipher machine and must be properly constructed with respect to the consonant and vowel alternation. This group is not directly encrypted with the cipher machine, but is specially constructed and sent as the first group of the ciphertext. The IV consists of five syllables[8] divided into three parts in the following order:

- 2 syllables - indicator;

[8]Two-letter syllable starting with a consonant followed by a vowel.

- 1 syllable - date;
- 2 syllables - message ID.

The indicator starts with a syllable, which tells that the message is encrypted with Condenser PBJ. Syllables in the range SA to ZU are reserved for this[9]. Due to possible errors in the telegraphic transmission, a second syllable in the same range is inserted (not necessarily the same as the first one). The following syllable is converted from the day of the message date. This information is used by the recipient to select the corresponding daily key before decryption.

As the IV consists of syllables only, numbers (date and message ID) have to be converted into letters first. The input numbers are always processed as two-digit numbers. If necessary, the number zero is inserted as padding before the input number ($1 \rightarrow 01$, $113 \rightarrow 01\ 13$, etc.). Each two-digit number x_1x_2 is converted into two letter syllable y_1y_2 using the following rule: if x_2 is in the range 1 to 5, use substitution Table 1, else use Table 2.

x_1/x_2	1	2	3	4	5	6	7	8	9	0
y_1	b	c	d	f	g	h	j	k	l	m
y_2	u	o	i	e	a	u	o	i	e	a

Table 1: IV number substitution A

x_1/x_2	1	2	3	4	5	6	7	8	9	0
y_1	n	p	q	r	s	t	v	w	x	z
y_2	u	o	i	e	a	u	o	i	e	a

Table 2: IV number substitution B

The last two syllables are converted from the message ID in the range 01 to 9999. If the message ID is two digits only, the first syllable is randomly selected, and the second syllable is calculated using tables 1 and 2 similarly as it was for the date. If the message ID is larger, it is divided into two parts and processed separately.

3.4 Alphabet

The letters of the alphabet must be mapped to the 10 tuples of the strips and rotors, in a scrambled order. As 26 letters cannot be mapped easily to

[9]The remaining syllables in the range BE to RU are reserved for other cipher types.

the 10 available positions, the letter Y is omitted. In the Czech language the Y is a vowel which can easily be replaced by the similar sounding I. The remaining letters are divided into 5 vowels (a, e, i, o, u) and 20 consonants. The 5 vowels appear twice on five of the strips and rotors: once in red and once in white/black. The 20 consonants are mapped to the 10 positions of the other five strips and rotors, by assigning two of them to each position: one printed in red and one in white/black. Figures (0-9) can be mapped to each strip and rotor, as there are exactly 10 positions.

3.5 Strips

Together with the rotors, the strips are the main secret parts of the machine. They are used for selection of the input characters and are situated at the top of the machine, where they are inserted into rectangular slots between the selectors. There are ten strips numbered 0-9. Each strip is used to encrypt a single input symbol, therefore ten strips are used in parallel on each step (computation cycle) to encrypt one 10-letter word. Five strips - marked with even numbers - are used for vowels (a, e, i, o, u). The remaining five strips are used for consonants, and are marked with odd numbers. Each strip has a rectangular shape with a pointed bottom end, which makes it easier to insert the strip into one of the slots. A strip holds ten tuples (see Figure 3). In the case of the consonant strips, a tuple consists of one number and two consonants. All the twenty available consonants are used and appear once on a consonant strip. In each consonant tuple, one consonant is white[10] and one is red. In the case of the vowel strips, a tuple consists of one number and one vowel. Because there are ten tuples, but only five vowels are used, each vowel appears twice on a vowel strip. Half of the vowels are white and half are red. The numbers in the tuples are used to encrypt numbers in the plaintext.

The colors are an important part, necessary in the process when the letters are set on the machine using the selectors. There is the following rule: when a consonant is set on a consonant strip, a vowel of the same color must be set on the right neighboring vowel strip. This results in five pairs of same-colored syllables in the block.

[10]For better visibility we have used black instead of white in the pictures.

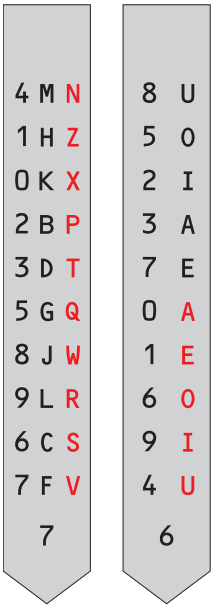


Figure 3: Strips 7 (consonants) and 6 (vowels)

Only one strip is fully described in the manual NARA (1924). However, because a long plaintext/ciphertext pair is available in the document, we were able to reconstruct the remaining strips (see Table 3). Note that on all vowel strips, the white letters are in the upper half of the strip and the red letters are in the lower half. On all consonant strips, the white letters are in the left column of the strip and the red letters are in the right column. This is probably done to simplify the use of the machine.

The reconstructed strips for the initial annual keys are in Table 3. Six additional annual keys, which were delivered with the cipher machine instructions, are in Figure 7 in the Appendix. The tuples are fixed (i.e. the same) on all strips, only their order is shuffled based on the annual key. For each strip, the order of the tuples determines their numeric value (used in encryption). The first (upper) tuple has a value of 0, the second one has a value of 1, etc., and the last one has a value of 9.

The daily key dictates the order in which the strips are inserted into the machine. Because of the strict structure of the formatted text, consonant strips must only be inserted into the consonant slots and vowel strips must only be inserted into the vowel slots.

In fact, the strip itself serves only for managing the position of the letter selection knob when entering text. According to (NARA, 1924), damaged or lost strips could be easily replaced with cardboard.

Value	7	6	5	0	3	2	9	4	1	8
0	4MN	8U	0KX	2I	5GQ	7E	3DT	3A	7FV	5O
1	1HZ	5O	2BP	7E	3DT	8U	8JW	2I	9LR	3A
2	0KX	2I	7FV	3A	6CS	5O	1HZ	7E	3DT	8U
3	2BP	3A	9LR	8U	7FV	2I	4MN	5O	1HZ	7E
4	3DT	7E	5GQ	5O	4MN	3A	9LR	8U	6CS	2I
5	5GQ	0A	3DT	6O	1HZ	9I	2BP	9I	0KX	1E
6	8JW	1E	6CS	0A	8JW	6O	7FV	4U	4MN	9I
7	9LR	6O	8JW	1E	0KX	4U	6CS	0A	2BP	4U
8	6CS	9I	1HZ	4U	9LR	0A	0KX	1E	5GQ	6O
9	7FV	4U	4MN	9I	2BP	1E	5GQ	6O	8JW	0A

Table 3: Reconstructed strips (annual key and daily key, derived from the example in the manual)

3.6 Rotors

At the front of the machine are ten rotors that are driven by the adding mechanisms inside the body. Together with the strips, they are the secret components of the machine that must be stored in a safe when the machine is not in use.

Each rotor has ten faces that correspond to the values 0-9. Each face contains a tuple that is identical to the tuple with the same value on the corresponding strip (see Table 3). Although the manual does not describe this, it is likely that due to space constraints, the tuples are printed vertically rather than horizontally as on the strips. This would explain why the rectangular windows are oriented vertically.

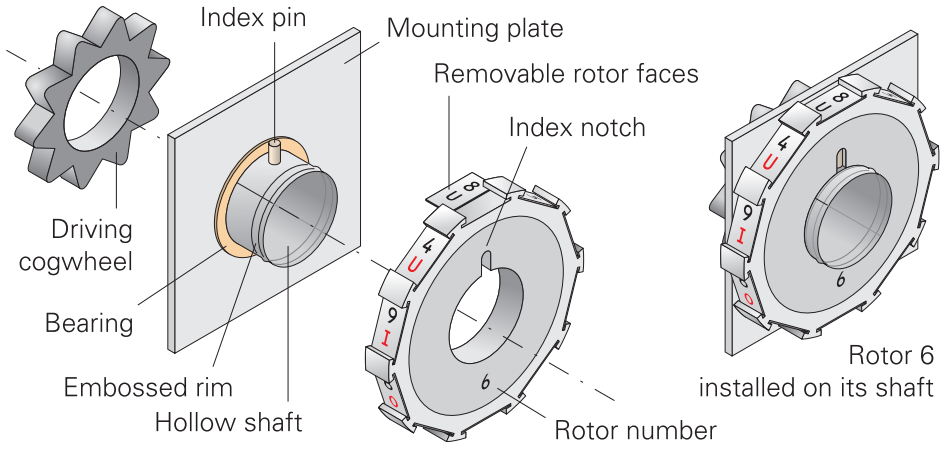


Figure 4: Educated guess of the rotor construction

Based on the manual (NARA, 1924), an educated guess about the construction of the rotors can be made, as shown in Fig. 4. Each rotor has its own construction. It consists of a vertical metal mounting plate with a bearing that holds a horizontal hollow shaft. One side of the shaft — we have assumed the right side — is for mounting the rotor. It has an index pin that mates with an index notch at the inner part of the rotor, to ensure that the rotor is mounted with the correct face up (i.e. the face that is visible through the window). It is likely that the rotor is driven by a cogwheel that is mounted to the same shaft, at the other side of the mounting plate. This cogwheel is then driven by the adding mechanism inside the body.

Note that the 10 faces can be removed from the rotor. This allows faces with new tuples to be installed as part of a new annual key. The tuples must be installed in the same order as on the corresponding strip.

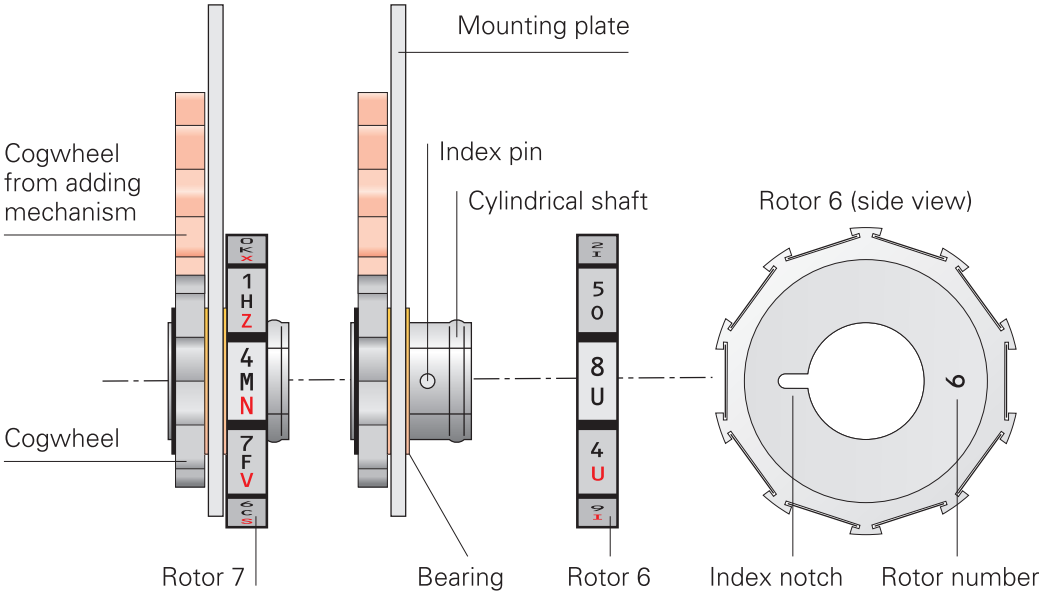


Figure 5: Top view of the rotor assembly, showing rotors 7 and 6 side-by-side

Before mounting the rotors onto their shafts, the reset lever must be engaged to ensure that the index pins are all pointing upwards. This corresponds to the neutral position of the rotor (i.e. the position that corresponds to a value of 0). It is likely that each rotor carried the same identification number as the strip with the identical tuples.

Fig. 5 shows the rotor assembly as seen from the top of the machine. Rotors 7 and 6 are shown here in their neutral positions, with the top faces (4MN and 8U respectively) visible through the windows in the rotor cover.

To ensure that the rotor does not come off when it rotates, part of the

hollow shaft is embossed. The manual describes that if the rotor falls off, the supplied *blunt awl* must be used to stretch the shaft somewhat. This is illustrated in Fig. 6.

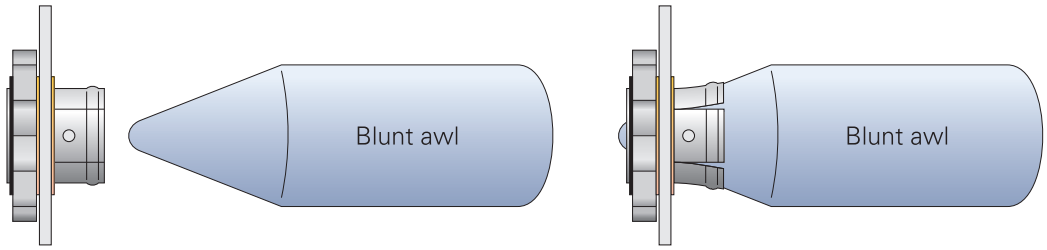


Figure 6: Stretching the rotor shaft with a blunt awl

4 Mathematical description

In this section, we summarize the encryption algorithm in a modern mathematical format and provide an initial cryptanalysis.

4.1 S-boxes

The machine is defined by 2 main substitutions (S-boxes), one for consonants κ, and one for vowels υ.

The consonant S-box maps 20 consonants from the set

$$C = \{B, C, D, F, G,\ H, J, K, L, M,\ N, P, Q, R, S,\ T, V, W, X, Z\}$$

to digits from $\mathbb{Z}_{10}$, as well as colors from the set $B = \{white, red\}$. Function

$$\kappa : C \to \mathbb{Z}_{10} \times B$$

is a bijective mapping. Given a consonant from C, we can produce a unique pair (digit, color), and vice-versa.

The vowel S-box maps 5 vowels from the set $V = \{A, E, I, O, U\}$, each in two colors (from B) to a digit from $\mathbb{Z}_{10}$. Again

$$\upsilon : V \times B \to \mathbb{Z}_{10}$$

is a bijective function that maps each unique pair (vowel, color) into a unique digit, and vice-versa.

Functions κ and υ can be combined (which we will denote by operator $\otimes$) to define a bijective function on syllables from $C \times V$ as follows:

$$(\kappa \otimes \upsilon)(c, v) = (x, y),\ \text{such that } \kappa(c) = (x, b),\ \text{and } \upsilon(v, b) = y.$$

This means that we join the numeric representation of the consonant c, with the numeric representation of the vowel v with the same color b as that of the consonant c (as assigned by κ).

4.2 Annual and daily keys

The annual key is a set of 10 permutations (bijections) on $\mathbb{Z}_{10}$, which we will denote by σ_i, with index $i \in \mathbb{Z}_{10}$. Even-numbered permutations influence vowels, odd-numbered permutations influence consonants.

The daily key is a specific permutation on $\mathbb{Z}_{10}$ that contains only odd numbers in even positions, and even numbers in odd positions. It can be constructed from two permutations ρ_0 (for vowels), ρ_1 (for consonants) on $\mathbb{Z}_5$ as follows:

$$\rho(i) = \begin{cases} 2\rho_1(i/2) + 1 & \text{if } i = 0 \mod 2 \\ 2\rho_0((i-1)/2) & \text{if } i = 1 \mod 2 \end{cases}$$

The configuration of the machine is determined by combining the annual key with the daily key as follows. Let i denote a (0-indexed) position in the group of letters. A "strip number" at position i is $\rho(i)$. Input number x in position i is mapped to output number y using permutation $\sigma_{\rho_i}^{-1}$. Note that we use inverse permutation here to be consistent with the original cipher description, where the annual key digits denote which number gets mapped to 0, 1, ...

4.3 Encryption

The plaintext is encoded into 2-letter groups (syllables, starting with a consonant and ending with a vowel), and padded to a size that is a multiple of 10. It can also contain digit pairs. The plaintext is then split into blocks of 10 symbols denoted by PT_i (indexed from 1). The first block of ciphertext CT_0 is prepared according to the rules from the manual (NARA, 1924).

Each block is encrypted as five 2-symbol groups (letters or digits). To encrypt 2-letter groups (syllables), we combine mapping $\kappa \otimes \upsilon$ with corresponding $\sigma_{\rho_i}^{-1}$ transformations. This defines 5 syllable encryption functions

$$Enc_j = (\sigma_{\rho_{2j}}^{-1}, \sigma_{\rho_{2j+1}}^{-1}) \circ (\kappa \otimes \upsilon)$$

Note that in practical use, when encrypting digits, the $(\kappa \otimes \upsilon)$ is replaced by identity. When encoding letters, the vowel color must match the consonant color (in a given pair).

The syllable decryption function is defined by

$$Dec_j = (\kappa \otimes \upsilon)^{-1} \circ (\sigma_{\rho_{2j}}, \sigma_{\rho_{2j+1}})$$

In practical use, $(\kappa \otimes \upsilon)^{-1}$ is omitted when the text indicates (by prefix NUME) that decryption should be numerical (digits). When decrypting letters (or producing ciphertext), the consonant is chosen in such a way that vowel and consonant colors are matching (in the given pair).

The block encryption works similarly to the modern CBC mode. First, we apply Enc_j in parallel to PT_i, and CT_{i-1}, respectively. We obtain 2 sequences of digits of length 10, $p_i \in \mathbb{Z}_{10}^{10}$, $c_{i-1} \in \mathbb{Z}_{10}^{10}$. These are summed together (modulo 10), and we get

$$c_i = p_i + c_{i-1}.$$

Finally, to produce ciphertext CT_i, we apply Dec_j in parallel to c_i.

In symbolic notation:

$$\begin{aligned} CT_i = Enc(PT_i) = Dec_0(Enc_0(PT_{i,0}, PT_{i,1}) + Enc_0(CT_{i-1,0}, CT_{i-1,1})), \\ \ldots, Dec_4(Enc_4(PT_{i,8}, PT_{i,9}) + Enc_4(CT_{i-1,8}, CT_{i-1,9})) \end{aligned}$$

Note that the encryption can be separated into 5 independent bigram-autokey ciphers applied in parallel. If the color mapping of consonants is known, it can be further separated into 10 independent substitution-autokey ciphers.

4.4 Key space

The machine configuration depends on the S-box settings. The consonant S-box is given as a permutation of 20 letters, with 20! options (ca. 2^{61}). The vowel S-box is given as a permutation of 10 letters, with 10! options (ca. 2^{22}). Note that in practice the white/red variants of vowels were continuous, which can decrease the uncertainty about the vowel S-box (if this fact is known). Thus, if the S-boxes are unknown, they contribute to the key space a factor of 2^{83} (at most).

The annual key is given by 10 permutations of size 10. Thus, the key space contribution of annual keys is at most $(10!)^{10}$, that is 2^{218}. Again, in practice this space was further reduced by some specific restrictions on the permutations that could be used. If the attacker knows these details, he could reduce the key space. E.g., in our research, we have found that vowel permutation had to keep vowels of the same color together. This rule reduces

the key space to $(10!)^5 \cdot (5! \cdot 5!)^5$, that is 2^{143}. This space is still too large to explore by exhaustive search even on modern computers.

Finally, the daily key is a combination of 2 permutations of size 5 (one for consonants, and one for vowels). Thus, the daily key contributes only $5! \cdot 5! = 14400$, which is approx. 2^{14} options.

4.5 Basic cryptanalysis

The studied algorithm combines a polyalphabetic substitution cipher (with a complex construction of the key) with an autokey (CBC-like mode).

We suppose that the analyst knows the algorithm, and may potentially have access to the S-boxes (e.g. by studying the machine). If the attacker gains access to the annual keys as well, modern computer analysis becomes trivial, as the daily key space is too small.

Another weakness of the cipher is its CBC-like autokey mode. If the attacker knows some blocks of plaintext, he can construct linear equations in the form $ct_i = pt_i + ct_{i-1}$ (in specific block positions). The unknowns in the equations are either syllables (100 unknowns), or can be divided into specific equations for consonants (20 unknowns) and vowels (10 unknowns).

An alternative to the key is a solution of this system of linear equations. Thus, if we have approximately 100 known P-C pairs, we can reconstruct mappings Enc_j between syllables and 2-digit numbers by simply solving the system with linear algebra.

The linear algebra based attack can also be implemented in the case when only 20 P-C pairs are known. First, we compute number equivalents of 20 consonants by solving a system of (at least) 20 equations with 20 unknowns. We should obtain 2 consonants for each digit. We then assign colors to the consonants, and use the vowels and their colors (same as the color of the preceding consonant) to prepare a system with 10 unknowns. It is not necessary to assign the colors for the consonants correctly, we can use any arbitrary color assignment. In the final step, we can compute digit assignment for vowels under the chosen assignments of colors, providing an equivalent key.

There are many remaining open questions, concerning the analysis. If we reconstruct equivalent keys (Enc_j) from known P-C pairs, how difficult is it to reconstruct the original machine settings and the annual key? How difficult is it to decrypt the ciphertext, if we don't have enough P-C pairs? How much data/computing power is needed for ciphertext-only attack?

Acknowledgements

This work was supported by grant VEGA 2/0054/24.

References

Eugen Antal and Pavol Zajac. 2023. The first Czechoslovak cipher machine. Cryptologia, 47:3, 239-260, DOI:https://doi.org/10.1080/01611194.2021.1998809.

AMZV. n.d. Personal files of Pavel Baráček-Jacquier. In Archiv Ministerstva zahraničních věcí České republiky, Praha (Archives of the Ministry of Foreign Affairs of the Czech Republic in Prague), f. Osobní spisy 1945-92, box 17, file Pavel Baráček-Jacquier.

AMZV. 1934. Document n. 16/II-B/34 - Memorandum o dnešním stavu šifrové služby ministerstva zahraničních vecí. In Archiv Ministerstva zahraničních věcí České republiky, Praha (Archives of the Ministry of Foreign Affairs of the Czech Republic in Prague), f. I. sekce 1918-39, oddělení I/1 a I/2 - běžná spisovna, box 16.

ITU. 1932. Telegraph Regulations Annex to the International Telecommunication Convention. International Radiotelegraph Conference. Madrid, 1932. Information available online: http://handle.itu.int/11.1004/020.1000/4.5.

NARA. 1924. Document n. 2251/Ic/24. - Instrukce k šifrovému stroji "Condenser PBJ" (Instructions for Cipher Machine Condenser PBJ). In National Archives and Records Administration, Record Group 457, Entry UD-22D 21, in the series: Archival and Historian's Source Files, box 4.

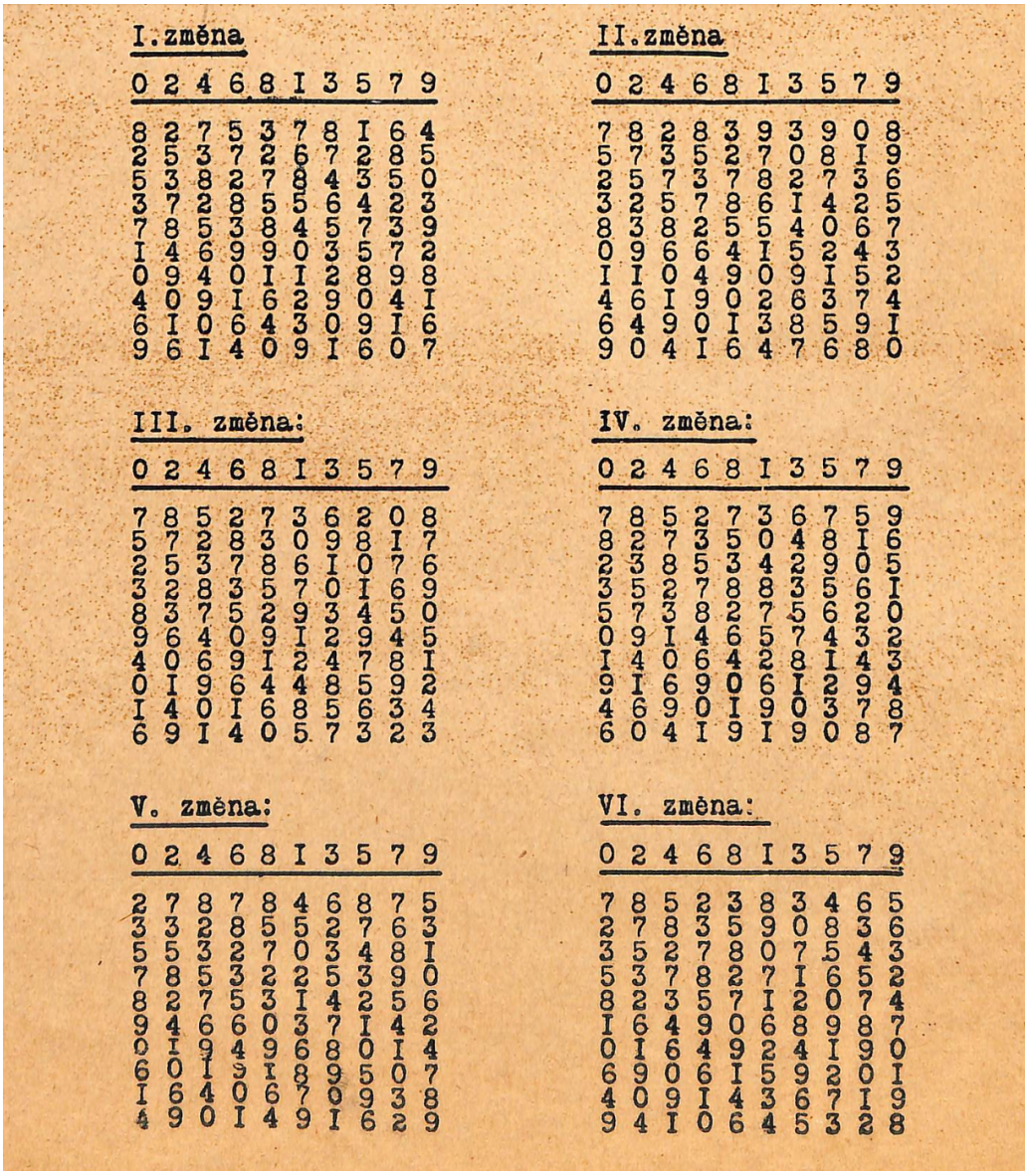
VUA. n.d. Military files of Pavel Baráček-Jacquier. In Vojenský ústrední archiv, Praha (Central Military Archives in Prague).

John McVey. 2013. Telegraphic code and message practice. Condensers. Information available online: https://www.jmcvey.net/cable/condensers/index.htm

Conrad Boe. 1937. The New Boe Code. Information available online: https://www.cryptomuseum.com/crypto/codebook/boe/

Ernest E. Peterson. 1929. Peterson International Code. Information available online: https://www.cryptomuseum.com/crypto/codebook/peterson/

Appendix A Annual keys



I.změna

0	2	4	6	8	I	3	5	7	9
8	2	7	5	3	7	8	I	6	4
2	5	3	7	2	6	7	2	8	5
5	3	8	2	7	8	4	3	5	0
3	7	2	8	5	5	6	4	2	3
7	8	5	3	8	4	5	7	3	9
I	4	6	9	9	0	3	5	7	2
0	9	4	0	I	I	2	8	9	8
4	0	9	I	6	2	9	0	4	I
6	I	0	6	4	3	0	9	I	6
9	6	I	4	0	9	I	6	0	7

II.změna

0	2	4	6	8	I	3	5	7	9
7	8	2	8	3	9	3	9	0	8
5	7	3	5	2	7	0	8	I	9
2	5	7	3	7	8	2	7	3	6
3	2	5	7	8	6	I	4	2	5
8	3	8	2	5	5	4	0	6	7
0	9	6	6	4	I	5	2	4	3
I	I	0	4	9	0	9	I	5	2
4	6	I	9	0	2	6	3	7	4
6	4	9	0	I	3	8	5	9	I
9	0	4	I	6	4	7	6	8	0

III. změna:

0	2	4	6	8	I	3	5	7	9
7	8	5	2	7	3	6	2	0	8
5	7	2	8	3	0	9	8	I	7
2	5	3	7	8	6	I	0	7	6
3	2	8	3	5	7	0	I	6	9
8	3	7	5	2	9	3	4	5	0
9	6	4	0	9	I	2	9	4	5
4	0	6	9	I	2	4	7	8	I
0	I	9	6	4	4	8	5	9	2
I	4	0	I	6	8	5	6	3	4
6	9	I	4	0	5	7	3	2	3

IV. změna:

0	2	4	6	8	I	3	5	7	9
7	8	5	2	7	3	6	7	5	9
8	2	7	3	5	0	4	8	I	6
2	3	8	5	3	4	2	9	0	5
3	5	2	7	8	8	3	5	6	I
5	7	3	8	2	7	5	6	2	0
0	9	I	4	6	5	7	4	3	2
I	4	0	6	4	2	8	I	4	3
9	I	6	9	0	6	I	2	9	4
4	6	9	0	I	9	0	3	7	8
6	0	4	I	9	I	9	0	8	7

V. změna:

0	2	4	6	8	I	3	5	7	9
2	7	8	7	8	4	6	8	7	5
3	3	2	8	5	5	2	7	6	3
5	5	3	2	7	0	3	4	8	I
7	8	5	3	2	2	5	3	9	0
8	2	7	5	3	I	4	2	5	6
9	4	6	6	0	3	7	I	4	2
0	I	9	4	9	6	8	0	I	4
6	0	I	9	I	8	9	5	0	7
I	6	4	0	6	7	0	9	3	8
4	9	0	I	4	9	I	6	2	9

VI. změna:

0	2	4	6	8	I	3	5	7	9
7	8	5	2	3	8	3	4	6	5
2	7	8	3	5	9	0	8	3	6
3	5	2	7	8	0	7	5	4	3
5	3	7	8	2	7	I	6	5	2
8	2	3	5	7	I	2	0	7	4
I	6	4	9	0	6	8	9	8	7
0	I	6	4	9	2	4	I	9	0
6	9	0	6	I	5	9	2	0	I
4	0	9	I	4	3	6	7	I	9
9	4	I	0	6	4	5	3	2	8

Figure 7: Six additional annual keys delivered with the instructions (NARA, 1924)

Appendix B Encryption example

To demonstrate the encryption process, we will explain the text formatting rules and IV construction, and encrypt the Czech example plaintext from (NARA, 1924). See also Sections 2 and 3.3.

Czech plaintext: *Vaše č.j. 2753: Experti přijedou do Prahy 18. března. Žádají přesné sdělení Vašeho programu pro porady. Zašlete obratem.*[11]

English raw translation: *Your num. 2753: Experts will come to Prague on March 18. They are asking for your agenda for briefings. Send immediately.*

In this example, we will use the initial annual key with the daily key $7, 6, 5, 0, 3, 2, 9, 4, 1, 8$ presented in Table 3.

Text formatting

The text should be formatted word-by-word from the left. Rule 1 is applied only to numbers. For most of the words, only rules 2 and 3 are checked. Rule 3 should be applied after rule 2. Rule 4 is applied only at the beginning and rule 5 only at the end. Words and abbreviations that do not add important meaning to the text may be dropped. This is not directly mentioned in the official rules, but is clear from the example in the document.

1. The first word is *vaše*. After checking rules 2-4 it can be directly written as *VASE* without change (except by removing the diacritical mark) because all the rules are satisfied. We also need to check the end of the text because of rule 4. An additional letter *U* is inserted at the end because the text ends with a consonant (*obratem*).

2. The next part is the message number. The number shortcut *č.j.* is dropped. For the number *2753* rule 1 is applied:

   (a) The *NUME* word is inserted.

   (b) *2753* consists of four digits, so the number 4 is inserted.

   (c) *2753* is inserted.

   (d) The checksum is inserted, $2 + 7 + 5 + 3 \mod 10 = 7$.

[11] In the original example word *zašlete* was changed during the encryption to *zašli* without any further explanation. Both forms of the word have the same meaning.

3. Word *experti* is changed to *EXUPERUTI*. Based on rule 3, a vowel u is inserted between two consonants (xp and rt). Rule 2 is not applied here.

4. Word *přijedou* is changed to *FIJEDOXU*. First, bigram *př* is changed to f based on rule 2, then consonant x is inserted between vowels *ou* based on rule 3.

5. Word *do* (Engl.: to) is dropped.

6. Word *Prahy* is changed to *FAHI*. Bigram *pr* is changed to f based on rule 2 and letter y is changed to i. Rule 3 is not applied here.

7. For the number *18* rule 1 is applied:

   (a) The *NUME* word is inserted.

   (b) *18* consists of two digits, so the number 2 is inserted.

   (c) *18* is inserted.

   (d) The checksum is inserted, $1 + 8 \mod 10 = 9$.

8. Word *března* is changed to *BUREZUNA*. Vowel u is inserted two times based on rule 3.

The remaining text is formatted in the same way. As it does not contain numbers, rule 1 is not applied anymore. Rule 2 is applied to the words:

- přesné, programu, pro, porady → fesne, fogramu, fo, poradi

Then rule 3 is applied to the words:

- fesne, sdělení, fogramu, zašli, obratem → fesune, sudeleni, foguramu, zasuli, oburatem

At this phase, we can start to form the groups. For each group, it is necessary to check rule 3 to ensure that there is no place in the text where two vowels or consonants are next to each other. Also, consonants could only be in odd places, and vowels only in even positions in the text.

- VASENUME42 | 7537EXUPER | UTI... → VASENUME42 | 7537XEXUPE | RUTI...; letter X was inserted before letter E in the second group because a vowel was located at an odd position.

- ... ZASULI | OBURATEMU → ZASULI | XOBURATEMU

Rule 5 is not applied, because the last group's length is 10.

The final formatted plaintext groups are:

PT_1	V	A	S	E	N	U	M	E	4	2
PT_2	7	5	3	7	X	E	X	U	P	E
PT_3	R	U	T	I	F	I	J	E	D	O
PT_4	X	U	F	A	H	I	N	U	M	E
PT_5	2	1	8	9	B	U	R	E	Z	U
PT_6	N	A	Z	A	D	A	J	I	F	E
PT_7	S	U	N	E	S	U	D	E	L	E
PT_8	N	I	V	A	S	E	H	O	F	O
PT_9	G	U	R	A	M	U	F	O	P	O
PT_{10}	R	A	D	I	Z	A	S	U	L	I
PT_{11}	X	O	B	U	R	A	T	E	M	U

IV construction

Let the day of the message date be 10, and the message ID 84.

- As an indicator the first two syllables are randomly chosen from the range SA to ZU $\rightarrow$ WO, SO.
- The day 10 is converted using Table 2 as NA.
- The Message ID has only two digits, so the first syllable is randomly chosen as LE. The number 84 is converted using Table 1 as KE.

The constructed IV is:

CT_0	W	O	S	O	N	A	L	E	K	E

Encryption

The encryption is performed sequentially block by block. In each computation cycle $i = 1, 2, \ldots$ the plaintext block PT_i is added to the previous ciphertext block CT_{i-1}, where $CT_0 = IV$. It can be directly calculated manually using Table 3.

One way is to process the PT/CT blocks by syllables. For each block i the following steps are performed:

- Split the CT_{i-1} block into five syllables

$$\{CT_{i-1,0}, CT_{i-1,1}\}, \ldots, \{CT_{i-1,8}, CT_{i-1,9}\}$$

  and the PT_i block into syllables

$$\{PT_{i,0}, PT_{i,1}\}, \ldots, \{PT_{i,8}, PT_{i,9}\}.$$

- For each tuple of $\{CT_{i-1,j}, CT_{i-1,j+1}\}/\{PT_{i,j}, PT_{i,j+1}\}$ syllable pair:
  - Find the value of consonant $CT_{i-1,j}$ in the strip at the position j, save as k_j.
  - Identify the color col_c of consonant $CT_{i-1,j}$.
  - Find the value of vowel $CT_{i-1,j+1}$ with the same color as col_c in the strip at the position $j+1$, save as k_{j+1}.
  - Find the value of consonant $PT_{i,j}$ in the strip at the position j, save as x_j.
  - Identify the color col_v of consonant $PT_{i,j}$.
  - Find the value of vowel $PT_{i,j+1}$ with the same color as col_v in the strip at the position $j+1$, save as x_{j+1}.
  - Calculate $y_j = x_j + k_j \mod 10$, $y_{j+1} = x_{j+1} + k_{j+1} \mod 10$.
  - Find the consonant with the value of y_j in the strip j as $CT_{i,j}$.
  - Find the vowel with the value of y_{j+1} in the strip $j+1$ as $CT_{i,j+1}$.

Please note that the colors of all the consonants are fixed (constants), and red vowels are only in the lower half of the vowel strips. This can speed up the encryption when the strips are used (e.g. Table 3). Also, identifying all values of k_j for ciphertext blocks separately from values x_j for plaintext blocks can speed up the process.

We will start with the first ciphertext block (IV), CT_0 = WOSONALEKE. In Table 3 the first consonant strip has id 7. The next (vowel) strip has id 6, etc.

- Letter W is in the 7th line, has red color and value 6.
- Letter O with the red color (the color of the previous consonant) is in the 8th line, and has value 7.
- Letter S is in the 7th line, has red color and value 6.

- Letter O with the red color is in the 6th line, and has value 5.
- Letter N is in the 5th line, has red color and value 4.
- Letter A with the red color is in the 9th line, and has value 8.
- Letter L is in the 5th line, has white color and value 4.
- Letter E with the white color is in the 3rd line, and has value 2.
- Letter K is in the 6th line, has white color and value 5.
- Letter E with the white color is in the 4th line, and has value 3.

The found characters are presented in Table 4 and are underlined.

Value	7	6	5	0	3	2	9	4	1	8
0										
1										
2								7E		
3										7E
4					4MN		9LR			
5				6O					0KX	
6	8JW		6CS							
7		6O								
8						0A				
9										
	k_0	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9
Value	6	7	6	5	4	8	4	2	5	3

Table 4: Numeric values of the first ciphertext block CT_0

We will process the first plaintext block, PT_1 = VASENUME42.

- Letter V is in the 10th line, has red color and value 9.
- Letter A with the red color is in the 6th line, and has value 5.
- Letter S is in the 7th line, has red color and value 6.
- Letter E with the red color is in the 8th line, and has value 7.
- Letter N is in the 5th line, has red color and value 4.
- Letter U with the red color is in the 8th line, and has value 7.

- Letter M is in the 4th line, has white color and value 3.
- Letter E with the white color is in the 4th line, and has value 2.
- Digit *4* is in the 7th line, has white color and value 6.
- Digit *2* with the white color is in the 5th line, and has value 4.

The found characters are presented in Table 5 and are underlined.

Value	7	6	5	0	3	2	9	4	1	8
0										
1										
2								7E		
3							4MN			
4					4MN					2I
5		0A								
6			6CS						4MN	
7				1E		4U				
8										
9	7FV									
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Value	9	5	6	7	4	7	3	2	6	4

Table 5: Numeric values of the first plaintext block PT_1

As a next step, values of y_i are calculated as $x_i + k_i \mod 10$ (Table 6).

i	0	1	2	3	4	5	6	7	8	9
x_i	9	5	6	7	4	7	3	2	6	4
k_i	6	7	6	5	4	8	4	2	5	3
$y_i = x_i + k_i \mod 10$	5	2	2	2	8	5	7	4	1	7
y'_i	G/Q	I	F/V	A	L/R	I	C/S	U	L/R	U

Table 6: First encrypted block CT_1

Then find the row marked y_i (in the Value column) in Table 3. The resulting letters (y'_i) are selected from the tuple at row y_i and column i. For vowels, we have only one possibility. The consonant is chosen in such a way that vowel and consonant colors are matching (in the given pair). Because in the y'_0, y'_1 pair vowel y'_1 is white, the white letter G is selected from the two

possibilities. Because in the y'_2, y'_3 pair vowel y'_3 is white, the white letter F is selected from the two possibilities. Because in the y'_4, y'_5 pair y'_5 is red, the red letter R is selected from the two possibilities, etc. The resulting CT_1 is GIFARICURU. The remaining blocks are encrypted in the same way.

The ciphertext groups are:

CT_1	G	I	F	A	R	I	C	U	R	U
CT_2	D	A	J	U	H	A	B	A	G	U
CT_3	H	I	F	A	R	U	F	E	F	U
CT_4	B	O	G	O	F	E	Q	E	N	E
CT_5	W	O	B	U	C	U	N	U	J	U
CT_6	J	I	N	I	V	I	R	A	W	E
CT_7	D	O	Z	A	H	O	R	O	V	O
CT_8	T	U	K	A	K	U	F	E	V	O
CT_9	V	U	R	U	D	O	Z	I	P	I
CT_{10}	J	E	Z	U	J	E	G	I	G	O
CT_{11}	C	O	M	E	N	A	Q	O	C	U