ADFGVX cipher

In cryptography, the **ADFGVX** cipher was a field cipher used by the German Army on the Western Front during World War I. ADFGVX was in fact an extension of an earlier cipher called **ADFGX**.

Invented by Lieutenant[1] Fritz Nebel (1891–1977)[2] and introduced in March 1918, the cipher was a fractionating transposition cipher which combined a modified Polybius square with a single columnar transposition.

The cipher is named after the six possible letters used in the ciphertext: `A`, `D`, `F`, `G`, `V` and `X`. The letters were chosen deliberately because they are very different from one another in the Morse code. That reduced the possibility of operator error.

Nebel designed the cipher to provide an army on the move with encryption that was more convenient than trench codes but was still secure. In fact, the Germans believed the ADFGVX cipher was unbreakable.[3]

Contents

Operation

For the plaintext message, "Attack at once", a secret mixed alphabet is first filled into a 5 × 5 Polybius square:

	A	**D**	**F**	**G**	**X**
A	b	t	a	l	p
D	d	h	o	z	k
F	q	f	v	s	n
G	g	i/j	c	u	x
X	m	r	e	w	y

`i` and `j` have been combined to make the alphabet fit into a 5 × 5 grid.

By using the square, the message is converted to fractionated form:

a	t	t	a	c	k	a	t	o	n	c	e
AF	AD	AD	AF	GF	DX	AF	AD	DF	FX	GF	XF

Next, the fractionated message is subject to a columnar transposition. The message is written in rows under a transposition key (here "CARGO"):

```
C A R G O
_________
A F A D A
D A F G F
D X A F A
D D F F X
G F X F
```

Next, the letters are sorted alphabetically in the transposition key (changing CARGO to ACGOR) by rearranging the columns beneath the letters along with the letters themselves:

```
A C G O R
_________
F A D A A
A D G F F
X D F A A
D D F X F
F G F   X
```

Then, it is read off in columns, in keyword order, which yields the ciphertext:

```
FAXDF ADDDG DGFFF AFAX AFAFX
```

In practice, the transposition keys were about two dozen characters long. Long messages sent in the ADFGX cipher were broken into sets of messages of different and irregular lengths to make it invulnerable to multiple anagramming.[3] Both the transposition keys and the fractionation keys were changed daily.

ADFGVX

In June 1918, an additional letter, `V`, was added to the cipher. That expanded the grid to 6 × 6, allowing 36 characters to be used. That allowed the full alphabet (instead of combining `I` and `J`) and the digits from `0` to `9`. That mainly had the effect of considerably shortening messages containing many numbers.

The cipher is based on the 6 letters ADFGVX. In the following example the alphabet is coded with the Dutch codeword 'nachtbommenwerper'. This results in the alphabet: NACHTBOMEWRPDFGIJKLQSUVXYZ. This creates the table below with the letters ADFGVX as column headings and row identifiers:

	A	**D**	**F**	**G**	**V**	**X**
A	N	A	1	C	3	H
D	8	T	B	2	O	M
F	E	5	W	R	P	D
G	4	F	6	G	7	I
V	9	J	0	K	L	Q
X	S	U	V	X	Y	Z

The text 'attack at 1200am' translates to this:

A	T	T	A	C	K	A	T	1	2	0	0	A	M
AD	DD	DD	AD	AG	VG	AD	DD	AF	DG	VF	VF	AD	DX

Then, a new table is created with a key as a heading. Let's use 'PRIVACY' as a key. Usually much longer keys or even phrases were used.

P R I V A C Y
A D D D D D A
D A G V G A D
D D A F D G V
F V F A D D X

The columns are sorted alphabetically, based on the keyword, and the table changes to this:

A C I P R V Y
D D D A D D A
G A G D A V D
D G A D D F V
D D F F V A X

Then, appending the columns to each other results in this ciphertext:
DGDD DAGD DGAF ADDF DADV DVFA ADVX

With the keyword, the columns can be reconstructed and placed in the correct order. When using the original table containing the secret alphabet, the text can be deciphered.

Cryptanalysis

ADFGVX was cryptanalysed by French Army Lieutenant Georges Painvin, and the cipher was broken in early June 1918.[4] The work was exceptionally difficult by the standards of classical cryptography, and Painvin became physically ill during it. His method of solution relied on finding messages with stereotyped beginnings, which would fractionate them and then form similar patterns in the positions in the ciphertext that had corresponded to column headings in the transposition table. (Considerable statistical analysis was required after that step had been reached, all done by hand.) It was thus effective only during times of very high traffic, but that was also when the most important messages were sent.

However, that was not the only trick that Painvin used to crack the ADFGX cipher.[3] He also used repeating sections of ciphertext to derive information about the likely length of the key that was being used. Where the key was an even number of letters in length he knew, by the way the message was enciphered, that each column consisted entirely of letter coordinates taken from the top of the Polybius Square or from the left of the Square, not a mixture of the two. Also, after substitution but before transposition, the columns would alternately consist entirely of "top" and "side" letters. One of the characteristics of frequency analysis of letters is that while the distributions of individual letters may vary widely from the norm, the law of averages dictates that groups of letters vary less. With the ADFGX cipher, each "side" letter or "top" letter is associated with five plaintext letters. In the example above, the "side" letter "D" is associated with the plaintext letters "d h o z k", and the "top" letter "D" is associated with the plaintext letters "t h f j r". Since the two groups of five letters have different cumulative frequency distributions, a frequency analysis of the "D" letter in columns consisting of "side" letters has a distinctively different result from those of the "D" letter in columns consisting of "top" letters. That trick allowed Painvin to guess which columns consisted of "side" letters and which columns consisted of "top" letters. He could then pair them up and perform a frequency analysis on the pairings to see if

the pairings were only noise or corresponding to plaintext letters. Once he had the proper pairings, he could then use frequency analysis to figure out the actual plaintext letters. The result was still transposed, but to unscramble a simple transposition was all that he still had to do. Once he determined the transposition scheme for one message, he would then be able to crack any other message that was enciphered with the same transposition key.[3]

Painvin broke the ADFGX cipher in April 1918, a few weeks after the Germans launched their Spring Offensive. As a direct result, the French army discovered where Erich Ludendorff intended to attack. The French concentrated their forces at that point, which has been claimed to have stopped the Spring Offensive.

However, the claim that Painvin's breaking of the ADFGX cipher stopped the German Spring Offensive of 1918, while frequently made,[5] is disputed by some. In his 2002 review of Sophie de Lastours' book on the subject, *La France gagne la guerre des codes secrets 1914-1918*, in the Journal of Intelligence History, (*Journal of Intelligence History*: volume 2, Number 2, Winter 2002) (https://web.archive.org/web/20060428005221/http://www.intelligence-history.org/jih/reviews-2-2.html) Hilmar-Detlef Brückner stated:

> Regrettably, Sophie de Lastours subscribes to the traditional French view that the solving of a German ADFGVX-telegram by Painvin at the beginning of June 1918 was decisive for the Allied victory in the First World War because it gave timely warning of a forthcoming German offensive meant to reach Paris and to inflict a critical defeat on the Allies. However, it has been known for many years, that the German *Gneisenau* attack of 11 June was staged to induce the French High Command to rush in reserves from the area up north, where the Germans intended to attack later on.
>
> Its aim had to be grossly exaggerated, which the German High Command did by spreading rumors that the attack was heading for Paris and beyond; the disinformation was effective and apparently still is. However, the German offensive was not successful because the French had enough reserves at hand to stop the assault and so did not need to bring in additional reinforcements.
>
> Moreover, it is usually overlooked that the basic version of the ADFGVX cipher had been created especially for the German Spring Offensive in 1918, meant to deal the Allies a devastating blow. It was hoped that the cipher ADFGX would protect German communications against Allied cryptographers during the assault, which happened.
>
> Telegrams in ADFGX appeared for the first time on 5 March, and the German attack started on 21 March. When Painvin presented his first solution of the code on 5 April, the German offensive had already petered out.

The ADFGX and ADFGVX ciphers are now regarded as insecure.

References

1. Friedrich L. Bauer: *Decrypted Secrets, Methods and Maxims of Cryptology*. Springer, Berlin 2007 (4. Aufl.), S. 173, ISBN 3-540-24502-2.
2. Friedrich L. Bauer: *Decrypted Secrets, Methods and Maxims of Cryptology*. Springer, Berlin 2007 (4. Aufl.), S. 53, ISBN 3-540-24502-2.
3. "Codes and Codebreaking in World War I" (https://web.archive.org/web/20100503103848/http://www.vectorsite.net/ttcode_04.html#m3). Archived from the original (http://www.vectorsite.net/ttcode_04.html#m3) on 3 May 2010. Retrieved 10 March 2010.

4. Newton, David E. (1997). *Encyclopedia of Cryptography*. Santa Barbara California: Instructional Horizons, Inc. p. 6.
5. "Painvin's manna had saved the French", wrote David Kahn, in *The Codebreakers - The Story of Secret Writing*, 1967, ISBN 978-0-684-83130-5, Chapter 9. Kahn also details the role that Painvin's decryption of German messages played in the French response to Operation Gneisenau.

Sources

- Childs, J. Rives, *General Solution of the ADFGVX Cipher System*, Aegean Park Press, ISBN 0-89412-284-3.
- Friedman, William F. *Military Cryptanalysis, Part IV: Transposition and Fractionating Systems*. Laguna Hills, California: Aegean Park Press, 1992.

External links

- A JavaScript implementation of the ADFGVX cipher (http://practicalcryptography.com/ciphers/adfgvx-cipher/#javascript-implementation)
- Another JavaScript implementation (http://peterhurford.com/other/adfgvx.html)
- A C implementation of the ADFGVX cipher (https://launchpad.net/adfgvx-tool)
