VỀ KỸ THUẬT NÂNG CAO ĐỘ BẢO MẬT MÃ AFFINE ỨNG DỤNG CHO VĂN BẢN TIẾNG VIỆT

TS. Nguyễn Đăng Minh
Trường Đại học Hòa Bình
Tác giả liên hệ: ndangminh@daihochoabinh.edu.vn

Ngày nhận: 10/3/2023
Ngày nhận bản sửa: 14/6/2023
Ngày duyệt đăng: 26/6/2023

Tóm tắt

Bài viết bàn về độ bảo mật của thuật toán affine và khả năng áp dụng thuật toán bảo mật trong thực tế. Những giải pháp nhằm nâng cao độ bảo mật của thuật toán, thêm khóa mã và áp dụng được cho văn bản tiếng Việt khi số ký tự trong bảng chữ cái vượt rất xa con số 26 trong bảng chữ cái, thường được thảo luận nhiều trong mã hóa thông tin.

Trình bày kết quả phần mềm thực hiện các giải pháp trên.

Phần mềm viết bằng ngôn ngữ Visual Basic98.

Từ khóa: *Affine, độ bảo mật, mã hóa văn bản tiếng Việt, phần mềm mã hóa sử dụng mã affine.*

Enhancing the Security of Affine Cipher for Vietnamese Text Encryption

Dr. Nguyen Dang Minh
Hoa Binh University
Corresponding author: ndangminh@daihochoabinh.edu.vn

Abstract:

This article discusses the security of the affine algorithm and its practical applications. It proposes solutions to enhance the security of the algorithm, including adding encryption keys and applying it to Vietnamese text, where the number of characters in the alphabet far exceeds the 26 characters in the English alphabet, which is commonly discussed in information encryption. The article also presents the results of software that implements these solutions, developed using Visual Basic98.

Keywords: *Affine, security, Vietnamese text encryption, affine encryption software.*

1. Đặt vấn đề

Thế giới bắt đầu thời kỳ chuyển đổi số. Thông tin không những được lưu giữ tại các máy tính cá nhân, mà còn trên các hệ thống lưu giữ mà không nhất thiết chỉ có chủ nhân mới thâm nhập được. Sự thâm nhập trái phép xảy ra thường xuyên không phải chỉ ở mức cá nhân, mà còn ở cấp cao hơn. Những thông tin quan trọng cần được giữ an toàn bằng việc biến đổi chúng (encription) thành dạng không hiểu được. Lý thuyết về mã hóa đã được phát triển và ứng dụng từ rất sớm, tuy vậy, trong thời đại ngày nay, vẫn cần phát triển để phù hợp với nhu cầu hiện tại.

Mã hóa bằng phương pháp affine [1, tr. 22] đã được biết đến và sử dụng từ khá lâu. Tuy vậy, ngày nay, không thể dùng được mã dưới dạng thuần túy vì tính bảo mật của mã quá thấp.

Thuật toán affine [1-4]:

Mật mã affine là một trường hợp đặc biệt của mật mã thay thế đơn ký tự nhưng tổng quát hơn. Khi dùng mã affine thuần túy, số lượng của chữ cái trong bảng chữ cái có kích thước m được gán một số từ phạm vi [0; m-1]. Sau đó, sử dụng số học modulo để đổi sang mỗi số tương ứng với một chữ cái khác. Trong bảng chữ cái gốc, số thứ tự là x, được chuyển sang một số mới, số thứ tự là y theo phép toán affine theo công thức: y=(ax+b) mod m. Ở đây, ta áp dụng phép toán số học modulo. A và m phải là các số nguyên tố cùng nhau.

2. Phân tích độ bảo mật của mã

Hàm mã hóa: Y=(ax+b) mod m, trong đó, x là số thứ tự của chữ cái ban đầu trong bảng chữ cái, m là số chữ cái trong bảng chữ cái, a là số nguyên tố cùng nhau với m, còn b là số bất kỳ, cuối cùng, y là số thứ tự trong bảng chữ cái đã mã hóa.

Yêu cầu nguyên tố cùng nhau giữa a và m đảm bảo cho mỗi chữ cái trong bảng cũ luôn chỉ có 1 chữ cái trong bảng mới và chỉ có thế, ta mới có thể tìm lại x khi biết y theo công thức sau: $x=a^{-1}(y-b)$. Trong đó, a^{-1} là nghịch đảo của a theo modul m. Nghĩa là, $a^{-1}a \bmod m=1$.

Ta có thể thấy các biến đổi khi mã hóa căn cứ vào các biến đổi trong bảng sau đây:

Cột "số ban đầu" là cột ghi số thứ tự của ký tự trong bảng chữ cái. Cột ghi "ký tự" là ký tự bảng chữ cái có số thứ tự bên trái. Cột thứ 3 và thứ 4 là các ký tự tiếp theo.

Như vậy, có thể thấy số 33 ứng với "!"; 54 ứng với "6". Sau khi dùng hàm mã hóa theo affine với lựa chọn a=3 và b=3, ta được bảng tiếp theo. Theo bảng đó, ký tự "!" biến thành "f", còn số "6" biến thành ¥, với a=13;b-3, ta có các chữ tương ứng "°" và "Á".

				Sau biến đổi y=mod(3x+3,256)				y=mod(13*x+3,256)			
số ban đầu	Ký tự	Số ban đầu	Ký tự								
33	!	54	6	102	f	165	¥	176	°	193	Á
34	"	55	7	105	i	168	¨	189	½	206	Î
35	#	56	8	108	l	171	«	202	Ê	219	Û
36	$	57	9	111	o	174	®	215	×	232	è
37	%	58	:	114	r	177	±	228	ä	245	õ
38	&	59	;	117	u	180	´	241	ñ	2	˥
39	'	60	<	120	x	183	·	254	þ	15	⁑
40	(	61	=	123	{	186	º	11	♂	28	
41	)	62	>	126	~	189	½	24	↑	41	)
42	*	63	?	129		192	À	37	%	54	6
43	+	64	@	132	„	195	Ã	50	2	67	C
44	,	65	A	135	‡	198	Æ	63	?	80	P
45	-	66	B	138	Š	201	É	76	L	93	]
46	.	67	C	141		204	Ì	89	Y	106	j
47	/	68	D	144		207	Ï	102	f	119	w
48	0	69	E	147	"	210	Ò	115	s	132	„
49	1	70	F	150	–	213	Õ	128	€	145	‘
50	2	71	G	153	™	216	Ø	141		158	ž
51	3	72	H	156	œ	219	Û	154	š	171	«
52	4	73	I	159	Ÿ	222	Þ	167	§	184	¸
53	5	74	J	162	c	225	á	180	´	197	Å

Nếu ta có "ABCDEFGH" thì với biến đổi affine với a=3,b=3 ta sẽ có dòng chữ như trên

Nghĩa là, cùng một ký tự, có thể biến thành các ký tự khác nhau nếu dùng a và b là những số khác nhau.

Khi a và b như nhau cho cả một văn bản được mã hóa thì các ký tự giống nhau trong văn bản gốc sẽ giống nhau trong văn bản đã mã hóa. Mã hóa như thế sẽ cho ta bảo mật không cao. Văn bản mã hóa dễ bị phá mã và phương pháp bảo mật affine gần như không được ứng dụng trong thực tế [2-4].

Để nâng cao độ bảo mật, phương pháp tốt nhất là mỗi ký tự trong văn bản, ta dùng một bộ a và b riêng. Chí ít là cho các ký tự liền nhau. Cách tạo bộ a và b như thế có thể tạo nên khóa mã (key word) cho phương pháp mã hóa.

Yêu cầu của key word dựa trên nguyên tắc bảo mật Kerckhoffs [2]:

1. The system must be practically, if not mathematically, indecipherable;

2. It should not require secrecy, and it should not be a problem if it falls into enemy hands;

3. It must be possible to communicate

and remember the key without using written notes, and correspondents must be able to change or modify it at will;

4. It must be applicable to telegraph communications;

5. It must be portable, and should not require several persons to handle or operate;

6. Lastly, given the circumstances in which it is to be used, the system must be easy to use and should not be stressful to use or require its users to know and comply with a long list of rules.

Điều 2 của nguyên tắc nêu rõ: "Nó không cần phải giữ bí mật, và nó sẽ không thành vấn đề nếu nó rơi vào tay kẻ thù (It should not require secrecy, and it should not be a problem if it falls into enemy hands)". Muốn vậy, key_word phải phụ thuộc vào nhiều tham số khác nhau nhưng vẫn phải đủ rõ để người áp dụng thuật toán không phải thực hiện quá nhiều thao tác hoặc phải nhớ thông tin.

3. Nâng cao độ bảo mật - mô phỏng, thuật toán

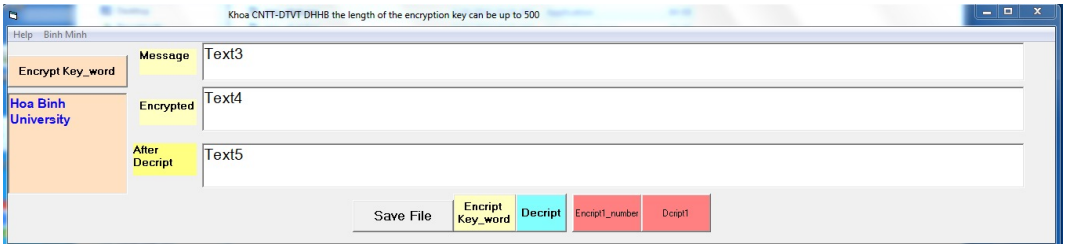
Keyword đương nhiên phụ thuộc từng ký tự của từ khóa, chiều dài của từ khóa, vị trí của ký tự trong văn bản cần mã hóa và chiều dài của văn bản cần mã hóa. Từ khóa không cố định về chiều dài nhưng đương nhiên, chỉ nên dùng các ký tự in được và không phải là ký tự điều khiển. Mọi sự phụ thuộc như trên có thể dễ thay đổi để mã hóa các loại văn bản khác nhau. Từ khóa với các tính năng như trên được thực hiện trên phần mềm đã được xây dựng và thực hành để đánh giá như dưới đây:

4. Phần mềm tìm số đảo cho He_so_a

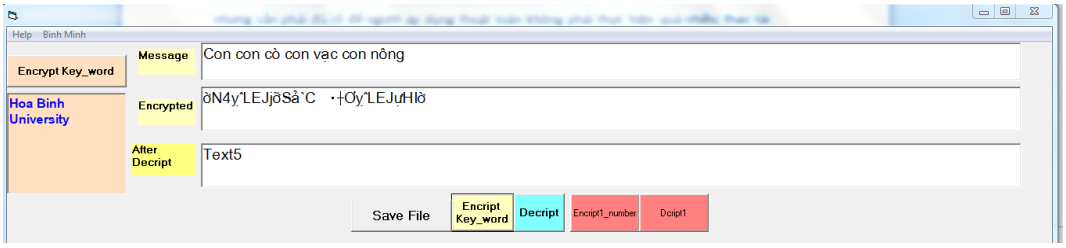
```
For i = 1 To So_m
If (He_so_a * i Mod So_m = 1) Then
Me.Caption = Str(i) + Me.Caption
Invert = i
End If
Next
```

5. Phần mềm tạo hệ số mã hóa khi có key word

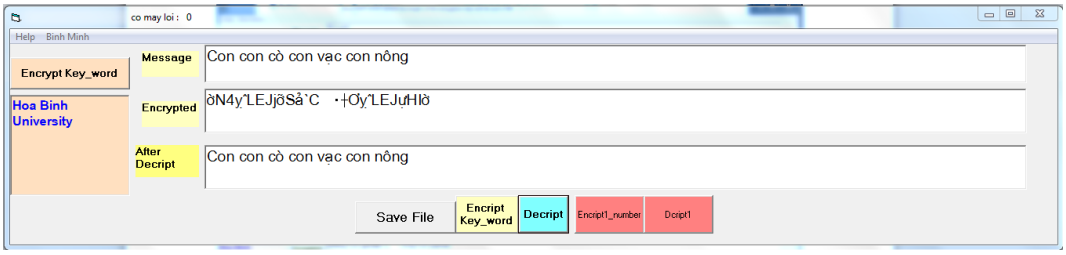
```
For i = 1 To Len(Van_ban)
j = i Mod 15
'Text6 = Text6 + Str(j)
so_b1 = So_b + j + 5 + (Len(Van_ban) Mod 30) + Key_ma(j) + 1
next
```



Muốn mã hóa, ta nhập vào ô ghi TEXT3, khi nhấn vào Encript, ta biến text3 thành text4 - thông tin đã mã hóa và kiểm tra việc mã hóa, giải mã bằng text5 khi nhấn vào decript.

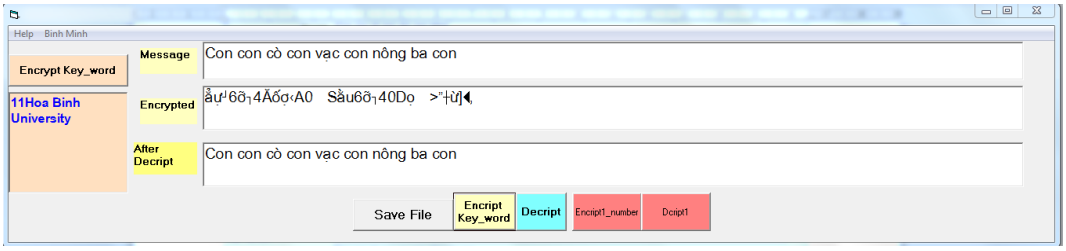


Và khi nhấn vào decript, ta được văn bản ban đầu trong text5.



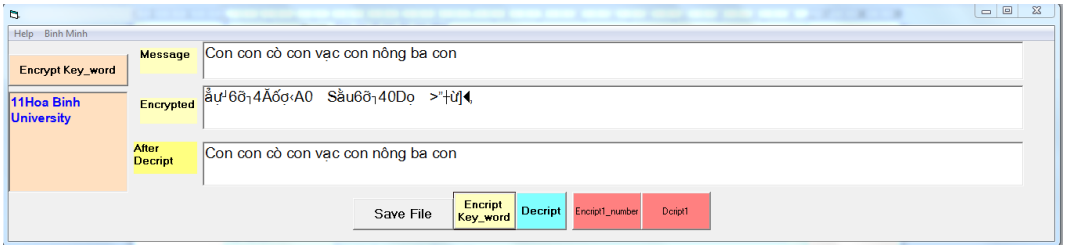
Trong ví dụ này, từ "con" được lặp lại nhiều lần, nhưng văn bản đã mã hóa không hề có sự lặp lại.

Bây giờ, hãy thực hiện một số thay đổi nhỏ:



Ta thêm vào từ "ba con" vào cuối dòng…Văn bản mã hóa bây giờ đã khác hẳn… như hình vẽ.

Bây giờ, ta thực hiện thay đổi nhỏ ở từ khóa:



Thêm số 12 vào trước trong từ khóa, văn bản mã hóa bây giờ đã khác hẳn.

Như thế, ngay cả khi phần mềm này rơi vào tay đối thủ và đối thủ nhận được văn bản mã hóa, cũng khó phát hiện ra văn bản ban đầu.

6. Kết luận

Bằng cách thêm key word, mã affine đã được nâng cấp đáng kể về độ bảo mật. Bảng chữ cái được mở rộng đến 256, do đó, phần mềm có thể tạo văn bản bảo mật cho một văn bản bất kỳ có trong máy tính. Phần mềm đang dùng để xử lý các bài text dùng font TCVN3 (ABC).

Tài liệu tham khảo

[1]. Nguyễn Xuân Dũng, *Bảo mật thông tin: Mô hình và ứng dụng*, NXB Thống kê.
[2] . http://www.crypto-it.net/eng/theory/kerckhoffs.html Kerckhoffs's principle Nguyên tắc Kirchopp
[3]. https://vi.wikipedia.org/wiki/M%E1%BA%ADt_m%C3%A3_Affine
[4]. https://cyberleninka.ru/article/n/statisticheskoe-issledovanie-affinnogo-shifra/viewer