PalArch's Journal of Archaeology of Egypt / Egyptology

CRYPTOGRAPHY OPTIMIZER AUTOLYSIS BIT GENERATOR TOOLS

Parvesh Kumar[1], Dr. K.P. Yadav[2]

[1]Phd Scholar Himalayan University India.
[2]Director Academic & research IIMT College of engineering (UP).

Parvesh Kumar, Dr. K.P. Yadav, CRYPTOGRAPHY OPTIMIZER AUTOLYSIS BIT GENERATOR TOOLS,-- Palarch's Journal Of Archaeology Of Egypt/Egyptology 17(15), 209-215. ISSN 1567-214x

ABSTRACT

In research "Cryptography Optimizer Autolysis bit Generator tools" method created and tried to fill the gaps between security layers, Data cryptography optimizer tools use table encrypted method for encrypt data and second table encrypted method for decrypt data files, here we use two table method for data encryption and decryption.

In the research we observe present world many types of tools uses to protect data and support various algorithm method and tables to protect the data and create a layer, dual layer and multi-layer to protect the data but there is various gaps between the data security. Hackers use various gaps method to decrypt the data.

Introduction

The number of cyber-attacks are increases day by day. In order to do so we need to know where their weaknesses are, before implementing measures to plug those gaps. The billions GB data are send or received from one network machine to other network machine so it is very necessary to protect data from the unauthorized access. The security protection is very critical issues in the software industry various types of algorithm table are used. Now present most of the tools support encryption or decryption method using one table (DES, AES, and RSA) and others. Cryptography techniques which are used to protect data are obtained from mathematical concepts and a set of rule based security calculations. Cryptography is the study of protect communications from outside sources. Encryption algorithms take the real data (plaintext) and converts it into (chiper text) coded.

In this research we have deeply studies about cryptography and find out the gaps between data security. Here in figure we have shown the encryption and decryption process.

In research "Cryptography Optimizer Autolysis bit Generator tools" method created and tried to fill the gaps between security layers, Data cryptography optimizer tools use table encrypted method for encrypt data and second table encrypted method for decrypt data files, here we use two table method for data encryption and decryption.

WHY WE NEED

FRADUSTERS

The involvement human contact, they will be vulnerable to fraud and misuse. Without proper monitoring in place, these business processes may be compromised.

OUTSIDERS RISK

The organizations' systems and employees, it is also vital to ensure that the third parties and vendors you work with have cyber security measures and policies in place. Organizations should put in place a regular and structured method to review and assess the security levels of these external parties to ensure that attackers are not able to exploit these loopholes to access the organization's network.

PEOPLE UNFAMILOUR

Worker organization's a weakest point, but also its greatest defense. A malicious staff may leak important information, or even allow attackers entry into the organization's network. An ignorant employee may even unknowingly leave an "open door" for attackers. However, an employee who is aware of the risks and educated about signs to look out for in a breach, is an organization's first line of defense. Ensure that employees are familiar with the risks and responses. All organizations, big and small, should be prepared for cyber breaches. Being able to identify and stop these attacks before they happen will save organizations from incurring substantial costs, and irreparable reputation damage.

NETWORK

The increased connectivity across devices and systems. A once isolated attack is a much more serious issue these days. Attackers may be able to find ways to enter a particular system through another "door" which may be easier to access. This is tough to monitor, and disconnecting devices and systems is not even an option as the world continues to evolve.

GAPS LACK

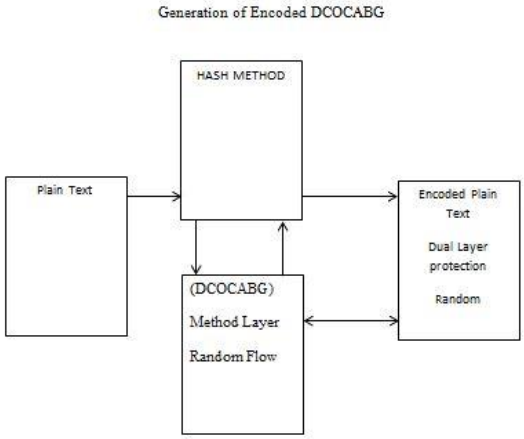
The number of cyber-attacks increase day to day. In order to do so we need to know where their weaknesses are, before implementing measures to plug those gaps.
During the webinar, we have observed the several gaps in cyber security that organizations face.

Cryptography Optimizer Autolysis Bit Generator tools (DCOCABG) table 1 encode the plain text and convert un-meaningful(UM) byte code, the converted UM code envelope using the DCOCABG method table.
DCOCABG encoding method 1 takes input as encoded plaintext advanced string code for decoding . It encapsulate it using methods1 for encapsulation further for sender.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1A	1B	1C	1D	1E	1F	GP	H0	IP	JP	KP	LP	MP	NP	P
2A	2B	2C	2D	2E	2F	GQ	H1	IQ	JQ	KQ	LQ	MQ	NQ	Q
3A	3B	3C	3D	3E	3F	GR	H2	IR	JR	KR	LR	MR	NR	R
4A	4B	4C	4D	4E	4F	GS	H3	IS	JS	KS	LS	MS	NS	S
5A	5B	5C	5D	5E	5F	GT	H4	IT	JT	KT	LT	MT	NT	T
6A	6B	6C	6D	6E	6F	GU	H5	IU	JU	KU	LU	MU	NU	U
7A	7B	7C	7D	7E	7F	GV	H6	IV	JV	KV	LV	MV	NV	V
8A	8B	8C	8D	8E	8F	GW	H7	IW	JW	KW	LW	MW	NW	W
9A	9B	9C	9D	9E	9F	GX	H8	IX	JX	KX	LX	MX	NX	X
0A	0B	0C	0D	0E	0F	YX	H9	IY	JY	KY	LY	MY	NY	Y
!A	ZB	ZC	ZD	ZE	ZF	ZX	H@	IZ	JZ	KZ	LZ	MZ	NZ	Z
@A	AO	CO	DO	EO	FO	G0	H%	$0	JA	0K	LN	MN	NN	NO
#A	AX	CX	DX	EX	FX	G1	H&	%0	JM	1K	LX	MX	NX	XO
$A	%	^	&	*	<	>	00	01	02	03	04	05	06	07

DCOCABG METHOD TABLE 1

Convert byte string with unique ID.
Converson of Cryptography Optimizer AutolysisBitGenerator tools (DCOCABG) method table 1.



DECRYPTION USE OF (DCOCABG)

To decrypt an (DCOCABG) encrypted ciphertext, it is necessary to undo each stage of the encryption operation in the reverse order in which they were applied. The three stage of decryption are as follows:

- Inverse Final Round
- AddRoundKey
- ShiftRows
- SubBytes
- Inverse Main Round

Of the four operations in (DCOCABG) encryption, only the AddRoundKey operation is its own inverse (since it is an exclusive-or). To undo AddRoundKey, it is only necessary to expand the entire (DCOCABG) key schedule (identically to encryption) and then use the appropriate key in the exclusive.

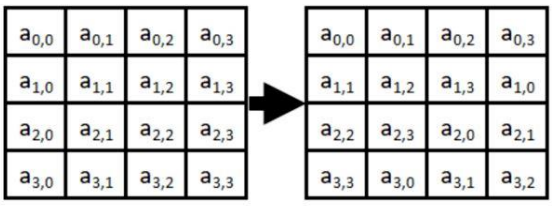
The other three operations require an inverse operation to be defined and used. The first operation to be undone is ShiftRows. The Inverse ShiftRows operation is identical to the ShiftRows operation except that rotations are made to the right instead of to the

left.
The next operation to be undone is the SubBytes operation. The Inverse S-Box is shown in the Table below. It is read identically to the S-Box matrix.

VARIENT (DCOCABG)
There are three variants of (DCOCABG) based on different key sizes (128, 192, and 256 \bits). Above we described the 64-bit version of the (DCOCABG) key schedule.All three variants of (DCOCABG) use a 64-bit block size, only the key sizes differ. The overall structure of the encryption stage of (DCOCABG) is identical for all three variants, but the number of rounds varies for the 128, 192, and 256-bit variants (10, 12, and 14 rounds respectively). The key schedule is different for each variant.

Encryption with (DCOCABG)
The encryption phase of (DCOCABG) can be broken into three phases the initial round, the main rounds, and the final round. All of the phases use the same sub-operations in different combinations as follows

- Initial Round
- AddRoundKey
- Main Rounds
- SubBytes
- ShiftRows
- ColumnsMix
- AddRoundKey
- Final Round



RESULTS
Key Generation of CRYPTOGRAPHY OPTIMIZER AUTOLYSIS BIT GENERATOR TOOLS (DCOCABG) or How to Generate the Key
we need to generate a key. With the help of this key we will encrypt the message. Now the interesting question is, how to generate the key, and where the key is to be used. Just follow the steps.

Step 1
Just select a random key of 10-bits, which only should be shared between both parties which means sender and receiver.
As I selected below
Select key:1010000010
Note:You can select any random number of 10-bits.

Step 2

Put this key into P.10 Table and permute the bits.

As I put key

Input	1	2	3	4	5	6	7	8	9	10
Output Should be	3	5	2	7	4	10	1	9	8	6

into P.10 Table.

Input	1	0	1	0	0	0	0	0	1	0
Output	1	0	0	0	0	0	1	1	0	0

Now the output will be
Key: 1000001100

Step 3
Divide the key into two halves, left half and right half;
{1 0 0 0 0} | {0 1 1 0 0}
Step 4
Now apply the one bit Round shift on each half:
Before round shift: {10000} | {01100}
After round shift: {00001} | {11000}
The output will be:
{0 0 0 0 1} {1 1 0 0 0}

Step 5:
Now once again combine both halve of the bits, right and left. Put them into the P8 table. What you get, that will be the K1 or First key.
Combine: 0 0 0 0 1 1 1 0 0 0

Step 7
Now just apply two round shift circulate on each half of the bits, which means to change the position of two bits of each halves.
left half: 00001
Right half: 11000
After the two rounds shift on each half out-put of each half will be.
Left half: 00100
Right half: 00011
Combine both together: As: 0 0 1 0 0 – 0 0 0 1 1

Step 8
Now put the bits into 8-P Table, what you get, that will be your second key. Table is also given in step 5. But here the combinations of bits are changed because of two left round shift from step 5. Check it in depth.

Combine bits: 0 0 1 0 0 0 0 0 1 1

Input	1	2	3	4	5	6	7	8	9	10
Combine-bits	0	0	1	0	0	0	0	0	1	1
Output Should be	6	3	7	4	8	5	10	9		
Output bits	0	1	0	0	0	0	1	1		

The output of the bits are your Second key or K2:

K2: 0 1 0 0 0 0 1 1

Finally we create both keys successfully

Bibliography / References

Tonouchi, T. (2001), Hyperbolic Domain Browser Implementation, available from http://www-dse.doc.ic.ac.uk/policies/software.html, October 2001.

Damianou, N., N. Dulay, E. Lupu and M. Sloman (2001). The Ponder Policy Specification Language. In Proceedings of the Policy Workshop 2001, HP Labs, Bristol, UK, Springer-Verlag, 29-31 January 2001

Corradi, A., N. Dulay, R. Montanari and C. Stefanelli (2001). Policy-Driven Management of Agent Systems. In Proceedings of the Policy Workshop 2001, HP Labs, Bristol, UK, Springer-Verlag, 29-31 January 2001.

Chomicki, J., J. Lobo and S. Naqvi (2000). A Logic Programming Approach to Conflict Resolution in Policy Management. In Proceedings of the 7th International Conference in Principles of Knowledge Representation and Reasoning, Breckenridge, Colorado, USA, Morgan Kaufmann Publishers, pp. 121-132, April 2000.

Epstein, Jennifer (October 13, 2009). Jaschik, Scott; Lederman, Doug (eds.). "Correcting a Style Guide". Inside Higher Ed. Washington, DC. Retrieved October 27, 2011.

APA Publications and Communications Board Working Group on Journal Article Reporting Standards (2008). "Reporting Standards for Research in Psychology: Why Do We Need Them? What Might They Be?" (PDF). American Psychologist. 63 (9): 839–851. doi:10.1037/0003-066x.63.9.839. PMC 2957094. PMID 19086746.

Kolter, J. Zico; Maloof, Marcus A. (December 1, 2006). "Learning to Detect and Classify Malicious Executables in the Wild". J. Mach. Learn. Res. 7: 2721–2744.

Tabish, S. Momina; Shafiq, M. Zubair; Farooq, Muddassar (2009). "Malware detection using statistical analysis of byte-level file content". Proceedings of the ACM SIGKDD Workshop on Cyber Security and Intelligence Informatics – CSI-KDD '09. p. 23.

December 31, 2016, at the Wayback Machine. Tom Meltzer and Sarah Phillips. The Guardian. October 23, 2009

IEEE Annals of the History of Computing, Volumes 27–28. IEEE Computer Society, 2005. 74 Archived May 13, 2016, at the Wayback Machine: "[...]from one machine to another led to experimentation with the Creeper program, which became the world's first computer worm: a computation that used the network to recreate itself on another node, and spread from node to node."

Top 10 Computer Viruses: No. 10 – Elk Cloner". Archived from the original on February 7, 2011. Retrieved December 10, 2010