

Forum:[Hack Forums](#)

Thread: [\[TUTORIAL\] Cryptography by Carrot | Learn the basics of modern cryptography/ciphers!](#)

CleanCarrot - June 21, 2018, 11:35 am

Hello everyone, ever wanted to learn some cryptography and decrypting basics? Here you are on the right place, you'll learn it in a progressive way.

INTRODUCTION

This part isn't the most important but it's a important part for understanding this tutorial. Cryptography is the art to hide messages and phrases. You'll need some informatic and mathematic knowledge to undersrtand this tutorial. Why does cryptography exists? To keep messages secret, but why would anyone keep them secret? For example to prevent from a war or just send someone a secret message. In this tutorial I will only talk about the "new" cryptography methods, the methods that are used in our society since the second war. I won't talk about decryption, I will soon make another tutorial about it. But to begin I'll still talk about the ceasar Cipher, the base of the whole cryptography...

1] Ceaser Cipher

The ceasar Cipher, named "rotation encryption" too is the easiest way to encrypt a message. It's named "ceasar Cipher" because ceasar himself used it (atleast it's said in the mythology haha). This cryptography method is based on alphabetic substitution:

Here ROT1 is used. A becomes B, B becomes C etc...
For example: Hello becomes ifmmp

Still today, this encryption type is used and often used for ROT13 (13 rotations). Decrypt messages encrypted with the ceasar cipher is very easy. You only need to do the inverse rotation to the

encrypt-rotation number and you get the clear text. A not so safe method to hide messages.

2] Vigenère cipher

This second encryption method I will show you is like the ceasar cipher (a letter becomes another letter) but this time, an "A" can for example become a "F" the first time it's written and a "T" the second time. For this encryption, a passphrase is used, you'll see it's a bit more interesting concerning security but still very easy to crack.

Okay imagine you want to crypt "hello" with the passphrase "Carrot".

You need to imagine:

Here you can see the passphrase "CARROT" and with the alphabet in line 1 and the table I just created you can find the encrypted "HELLO", here "JECCC".

With every Character of the password, a new alphabet is started at position "1" from this character. For example "B" -> "B-C-D....."

This passphrase is repeated as many times as there are characters in the message that has to be encrypted. With the normal alphabet at the top of the table, you can easily encrypt your message with the passphrase you chose.

Even if it's a bit stronger than the ceasar cipher, it's still very easy to decrypt this encryption method.

1] The symetric-key algorithm system

To understand this part you'll need to understand what an encryption function is:

Code $f : M \rightarrow C$

Where M is the whole clear message and C the whole encrypted message. This means that by using the function f on the clear message, you'll get the encrypted message. This applies to all symetric-key algorithms.

Symetric-key algorithm: A algortihm that uses the same key for the encryption and for the decryption.

To decrypt the message crypted with f:

Code $f^{-1} C \rightarrow M$

This means that you can't get two different clear messages with one encrypted message. To decrypt the encrypted message, you'll need the inverse function of the function you used to encrypt the clear message. That makes the decryption very easy if you know the key.

2] The Vernam Cipher

1917, Vernam invents an encryption algorithm that was so secured, that it was even used to encrypt the messages between the USA and the URSS during the cold war.

How to use :The key has the same size as the message to encrypt

The characters of this key are generated randomly

The key can't be used more than one time

How it works:

First of all the message, currently "hello" is converted to binary :

Code Hello: 01001000 01100101 01101100 01101100 01101111

(read this tutorial if you have no idea how to do this : https://en.wikipedia.org/wiki/Binary_code)

Then the key is generated. Lets imagine the key is "Carro"

Code Carro: 01000011 01100001 01110010 01110010 01101111

Then you need the logical operator XOR to encrypt this message with the help of the key.

XOR is used like this:

Code 1 XOR 1 = 0

1 XOR 0 = 1

0 XOR 0 = 0

0 XOR 1 = 1

The encryption algorithm is this:

Code E-> M XOR K

With K the key, here "Carro", M the message, here "hello" and E the encrypted result.

It works like this:

Code M: 01000011 01100001 01110010 01110010 01101111

K: 01001000 01100101 01101100 01101100 01101111

E: 00001011 00000100 00011110 00011110 00000000

The encrypted message is "00001011 00000100 00011110 00011110 00000000"

To decrypt you take again the encryption key, here "Carro" and you apply the same algorithm with the same key to the encrypted message:

Code M = K (*)⁽⁻¹⁾ E

Here, (*) is the encryption algorithm that is used: here XOR

(*)⁽⁻¹⁾ is the inverse function of the encryption algorithm used: here the inverse function of XOR is still XOR (cool ^^)

So we can decrypt E with K using XOR

Code E: 00001011 00000100 00011110 00011110 00000000

K: 01001000 01100101 01101100 01101100 01101111

M: 01000011 01100001 01110010 01110010 01101111

Okay, now we finished the easy part haha

Quote:Whaaaat? This was the easy part?

Yeah, this was the easy part haha, now we'll talk about the...

3] The FEISTEL SCHEMA

Okay we now know how to encrypt a string with a key that has the size of the string. But how about a short key, but still with optimal security? Therefore, some searches found a way to encrypt strings with 80 to 120 bits keys.

How does it work?

Okay let's imagine we take the key "0010" and we want to encrypt the message "0110 0001" with it. our argument for the function f is "0010", our key.

We take the number of bits in the key, here $n=4$, and we split our message into blocks that are $2n$ bits large.

Here "0110 0001" is already 8 bits, so we won't need to split it.

This block we split it into two other blocks with n bits. Here our message is splitted into:

Code G: 0110

D:0001

(G and D are variables)

with this two blocks we will generate two encrypted blocks containing each n bits:

In this schema you can see the working of the FEISTEL Schema.

the left encrypted block here, the block "L" contains the data from our right not-encrypted block, "D"
The right encrypted block contains $G \text{ XOR } f(D)$

That means:

Code D = 0001

K = our key, here 0010

$f(D) = D \text{ XOR } K$

$f(D) = 0001 \text{ XOR } 0010 = 0011$

and $G \text{ XOR } f(D) = 0110 \text{ XOR } 0011 = 0101$

Our encrypted message is "0001 0101".

Here you can see that the left part is the same as the right part of the clear message, but to prevent this, we create a loop, looping this FEISTEL Schema. For example the DES encryption algorithm uses this loop 16 times.

The f function varies between the different encryption methods, I was quite simple I just used the $Fx \rightarrow K \text{ XOR } x$ function.

Normally, new keys are generated from the official key every loop, this is named the "key schedule" :

example K1, K2, K3 ... K16 for 16 loops.

To decrypt a message crypted with this schema it's quite easy, you just need to begin from the encrypted message and do the same loops but beginning with the K16 if you had 16 loops.

Many modern cryptography use the Feistel cipher like: AES

DES

Blowfish

ICE

And many other

Conclusion

So, this mini tutorial for beginners is finished, but don't worry here is just the basis of cryptography I will do some other tutorials if I get some nice feedback haha! I hope you enjoyed and feel free to ask me any questions or to correct some little details if you find errors!

Thanks for reading!

Cheers ?

Ren And Stimpy - June 21, 2018, 11:36 am

Holy shit! Nice Thread!

Posting to come back later for the read.

CleanCarrot - June 21, 2018, 11:37 am

(Jun 21, 2018 11:36 AM)Ren And Stimpy Wrote: [\[CITE\]](#) Holy shit! Nice Thread!

Posting to come back later for the read.

Haha cool^^ I will provide some more tutorials I think

Path - June 21, 2018, 11:51 am

very informative, you been pumping so many good threads at the moment, keep up the good work!

CleanCarrot - June 21, 2018, 11:53 am

(Jun 21, 2018 11:51 AM)Path Wrote: [\[CITE\]](#) very informative, you been pumping so many good threads at the moment, keep up the good work!

Thanks! Yeah I will do it haha ^^

Ben Stiller - June 21, 2018, 1:23 pm

Oh !

Its a great tutorial !

Have you ever thought of setting up writing services? For homework, work etc. ?
Edit : Oops, my bad you have already open a service like this ! GLWS btw !

Nice :D

CleanCarrot - June 21, 2018, 1:34 pm

(Jun 21, 2018 01:23 PM)Ben Stiller Wrote: [\[CITE\]](#) Oh !
Its a great tutorial !

Have you ever thought of setting up writing services? For homework, work etc. ?
Edit : Oops, my bad you have already open a service like this ! GLWS btw !

Nice :D
Haha thanks ^^
Yeah I need some cash hehe

? ?

CleanCarrot - June 22, 2018, 3:06 pm

I will soon release a tutorial using autoIT!! Still about cryptography.
Cheers

?Stark - June 22, 2018, 10:36 pm

Nice thread
I sent you PM about help :)

CleanCarrot - June 23, 2018, 2:20 am

(Jun 22, 2018 10:36 PM)?Stark Wrote: [\[CITE\]](#) Nice thread
I sent you PM about help :)
Thanks ^^

CleanCarrot - June 24, 2018, 10:24 am

Hey, if anyone is really interested about cryptography please send me a PM, I search a partner!

ApexLead - June 25, 2018, 5:37 am

Very good job mate, posting to come back later and expand my knowledge :)

IzzyMichiel - June 25, 2018, 7:02 am

Dude... Nice tutorial!
Thanks for sharing this with the community.

Thanks,

CleanCarrot - June 25, 2018, 9:37 am

(Jun 25, 2018 07:02 AM)IzzyMichiel Wrote: [\[CITE\]](#) Dude... Nice tutorial!
Thanks for sharing this with the community.

Thanks,
You are welcome ^^

niksan258 - June 29, 2018, 2:39 pm

Real nice of someone taking his time to explain to the new people :D Good job mate !

CleanCarrot - July 1, 2018, 4:55 am

(Jun 29, 2018 02:39 PM)niksan258 Wrote: [\[CITE\]](#) Real nice of someone taking his time to explain to the new people :D Good job mate !

Thanks a lot ^^

?elevision - July 2, 2018, 10:46 am

Nice work. This is a great tutorial. Thanks.

Last_viking - July 2, 2018, 11:09 am

What a tutorial your tutorial simply the best easy to understand keep good work up Ma man

MorsE - July 3, 2018, 6:55 am

Great Info, Thanks for sharing! ?

CleanCarrot - July 9, 2018, 4:56 am

(Jul 3, 2018 06:55 AM)MorsE Wrote: [\[CITE\]](#) Great Info, Thanks for sharing! ?

You're welcome ^^

mrusafreak - July 9, 2018, 5:09 am

This is a good tutorial !

CleanCarrot - July 11, 2018, 10:44 am

(Jul 9, 2018 05:09 AM)mrusafreak Wrote: [\[CITE\]](#) This is a good tutorial !

Thanks's a lot :)

salvui - October 23, 2018, 1:12 am

Congratulations for the tutorial, explained very well and understandable by everyone

CleanCarrot - October 28, 2018, 6:27 am

(Oct 23, 2018 01:12 AM)salvui Wrote: [\[CITE\]](#) Congratulations for the tutorial, explained very well and understandable by everyone

Thanks ^^

janulark - October 28, 2018, 6:16 pm

Very informative guide. Thank you very much

CleanCarrot - November 14, 2018, 1:50 pm

(Oct 28, 2018 06:16 PM)janulark Wrote: [\[CITE\]](#) Very informative guide. Thank you very muchYou're welcome :)

CleanCarrot - August 19, 2021, 6:15 am

What would you all say about a part 2 of these tutorials? Actually preparing one about MD5 and bloc encryption...

Jimmy00 - October 26, 2021, 5:46 pm

nice thread brother good read