

Forum:[Hack Forums](#)

Thread: [Circular 4-bit rotations \(for fun\)](#)

0xDEAD10CC - May 27, 2017, 4:34 pm

I came up with the idea of circular rotations on each box of 4 bits (counter-clockwise in this case) earlier today and decided to try and write a function that would save the values back to the 2 16-bit integers. It was a pretty fun thing to write :)

Code #include <stdio.h>

#include <stdint.h>

```
void rotate_4b(uint16_t *v1, uint16_t *v2)
{
    int n = 0;
    while (n < 16)
    {
        int swap_v1 = (((*v1 >> n) & 1) << 1) | ((*v2 >> n) & 1);
        int swap_v2 = ((*v1 >> n) & 2) | ((*v2 >> (n + 1)) & 1);
        *v1 = (*v1 & ~(3 << n)) | (swap_v1 << n);
        *v2 = (*v2 & ~(3 << n)) | (swap_v2 << n);
        n += 2;
    }
}
```

```
int main(void)
{
    int i = 0;
    uint16_t v1 = 0x0A0A;
    uint16_t v2 = 0xB0B0;

    for (; i < 4; ++i)
    {
        rotate_4b(&v1, &v2);
        printf("0x%.4x\n", v1);
        printf("0x%.4x\n\n", v2);
    }
}
```

return 0;

Code /*
before:

0001 0010 0011 0100 = 0x1234
0101 0110 0111 1000 = 0x5678

after:

0111 0100 0111 1000 = 0x7478
0000 0011 0011 0100 = 0x0334

```
0 <- 1
| |
1 -> 0
```

= ...

```
1 - 0
| |
0 - 1
```

*/
Section is not that active, so I'll post this even though it's not terribly useful.

xCh3rx - May 28, 2017, 4:59 am

Nice fun script :) Are you using gvim as text editor?

stackoverflowin - May 28, 2017, 5:03 am

Code VOID fnRotateBits(PUINT16 pu16First, PUINT pu16Second)

```
{
    for (UINT8 i = 0; i < 16; i += 2, *pu16First = (*pu16First & ~(3 << i)) | ((((*pu16First >> i) & 1) << 1) | ((*pu16Second >> i) & 1)) << i), *pu16Second = (*pu16Second & ~(3 << i)) | (((*pu16First >> i) & 2) | ((*pu16Second >> (i + 1)) & 1) << i));
}
```

Was bored. Kinda took it the way you did it and re-wrote it, just waiting for someone to arrive. Note: for meme purposes only.

0xDEAD10CC - May 28, 2017, 4:44 pm

(May 28, 2017 04:59 AM)xCh3rx Wrote: [\[CITE\]](#) Nice fun script :) Are you using gvim as text editor? Yeah that is gvim.

Ronnie Coleman - September 10, 2017, 6:32 am

Can you explain what you mean by circular 4-bit rotation? I don't understand the example. My intuition was that each 4-bit block is circularly shifted to the left (which is not the case).

Nvm I get it now.

Annabelle Lee - September 10, 2017, 8:02 am

(Sep 10, 2017 06:32 AM)_Russ Wrote: [\[CITE\]](#) Can you explain what you mean by circular 4-bit rotation? I don't understand the example. My intuition was that each 4-bit block is circularly shifted to the left (which is not the case).

Nvm I get it now.

Can you please explain it then? I've read it a few times and still have trouble grasping what he intended.

0xDEAD10CC - September 10, 2017, 12:25 pm

I thought it was pretty simple to understand:

^ For each "group" of 4 bits aligned up like so from 2 numbers, exchange the bits like above. In this case there are still 3 more rotation groups to the right of the left 2 bits between each of the 2 numbers.

I imagine this could be used as some kind of encryption or cipher algorithm if you had a key which determined the mis-alignments of the bits and how many rotations to do at each mis-alignment.

Annabelle Lee - September 10, 2017, 6:31 pm

(Sep 10, 2017 12:25 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I thought it was pretty simple to understand:

^ For each "group" of 4 bits aligned up like so from 2 numbers, exchange the bits like above. In this case there are still 3 more rotation groups to the right of the left 2 bits between each of the 2 numbers.

I imagine this could be used as some kind of encryption or cipher algorithm if you had a key which determined the mis-alignments of the bits and how many rotations to do at each mis-alignment. Oh!!! Okay, that diagram made it a LOT clearer. Thanks. That's pretty nifty.

Phizo - September 10, 2017, 6:42 pm

(Sep 10, 2017 12:25 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I imagine this could be used as some kind of encryption or cipher algorithm if you had a key which determined the mis-alignments of the bits and how many rotations to do at each mis-alignment.

Not in isolation. A cipher defined by linear operations is trivially broken with linear cryptanalysis. See: <https://hackforums.net/showthread.php?tid=5678742>

The only time such an operation is acceptable is when the cipher isn't defined by the linear operation. This is typically done using S-boxes, as in DES, which provides the core security of many ciphers by providing non-linearity. A linear encryption function is of the form $E(X \oplus Y) = E(X) \oplus E(Y)$. When a cipher can be expressed as such, it is trivially broken given $n + 1$ plaintext-ciphertext pairs. Bit rotation is not any more secure than any other linear bitwise operation. No more secure than the bitwise NOT in DES's Feistel function when the eight 6-bit S-boxes are removed.

Note that using a key (even though one of the blocks used in parallel as you do is essentially a multi-byte key) makes no difference to this. This is essentially a polyalphabetic cipher, like the Vigenère cipher, and what I assume you mean by 'key', a piece of information which shifts the bits, makes no difference, as that is essentially Caesar's cipher, which the Vigenère cipher already extends. Such ciphers are trivially broken.

Phizo Wrote: [\[CITE\]](#) As for how trivial it is to recover the key given XOR-encrypted ciphertext, it is the same as breaking a polyalphabetic substitution cipher. You determine the period p (length of the key) using the Index of Coincidence, then group each ciphertext character $c[n \cdot p + i]$ for all n, i such that $0 \leq n < \text{length}[c] / p$ and $0 \leq i < p$, and then solve each group as a monoalphabetic substitution cipher, which is trivial with frequency analysis. This is introductory cryptanalysis. You can only do this if $\text{length}[k] \leq \text{length}[c] / 2$ for a key k , though, which is typical. Otherwise, only by brute-force.

0xDEAD10CC - September 10, 2017, 7:12 pm

I never claimed at all that it would be secure just the same as I would never recommend simple xor encryption safe, which is why my statement about it doesn't read that way; if you re-read what I actually said, I only mentioned it being used as a potential way to manipulate data but I never made any claim towards the level of security it would provide. But it's another unique way of manipulating data. I still think you misread what I was thinking about doing...

Code 111[10]000
000[01]111

111[01]000

[00]010000

11110000
00010000

Phizo - September 10, 2017, 7:14 pm

(Sep 10, 2017 07:12 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I never claimed at all that it would be secure just the same as I would never recommend simple xor encryption safe, which is why my statement about it doesn't read that way; if you re-read what I actually said, I only mentioned it being used as a potential way to manipulate data but I never made any claim towards the level of security it would provide. But it's another unique way of manipulating data.

Your post does not read that way. It was not indicative at all. It's good to be explicit about such things for everyone's sake. The same way you will frequently respond to someone who has not been explicit about some C++ semantics or code.

0xDEAD10CC - September 10, 2017, 7:16 pm

(Sep 10, 2017 07:14 PM)Phizo Wrote: [\[CITE\]](#) Your post does not read that way. It was not indicative at all. It's good to be explicit about such things for everyone's sake. The same way you will frequently respond to someone who has not been explicit about some C++ semantics or code.

If it wasn't indicative then why did you assume I claimed any decent level of security against such an algorithm? That's not the proper logical reasoning you should be using if someone provides inadequate data of any kind for attempting to comprehend and analyse someone's post. I didn't really intend on going too much in detail about my idea because I never assumed people would take it seriously enough to follow and use in any real-world scenario, which is why the details were intentionally left as vague. Nonetheless - My idea was as follows:

Code 111[10]000
000[01]111

111[01]000
[00]010000

11110000
00010000

Where the number of shifts for the alignment of the 2 numbers before each 4 bit rotation would be stored in some way within a key for the encryption and decryption.

Phizo - September 10, 2017, 7:19 pm

(Sep 10, 2017 07:16 PM)0xDEAD10CC Wrote: [\[CITE\]](#) If it wasn't indicative then why did you assume I claimed any decent level of security against such an algorithm? My idea was as follows:

Code 111[10]000
000[01]111

111[01]000
[00]010000

11110000
00010000

I know very well what your idea was. It is suggested by you prompting its use "as" an encryption algorithm. Not "in." Semantics are very important when conveying ideas. Regardless, I was just saying for everyone, just as you do when you attempt to take someone's post and make it more explicit frequently. I'm not sure why you feel the need to argue, as there is nothing to argue.

(Sep 10, 2017 07:16 PM)0xDEAD10CC Wrote: [\[CITE\]](#) Where the number of shifts for the alignment

of the 2 numbers before each 4 bit rotation would be stored in some way within a key for the encryption and decryption.

Security through obscurity. Not a good principle in general. This has no benefit over traditional bitwise like XOR, which is actually a much better option because XOR is an involutory function (i.e. it is its own inverse: $f(f(x)) = x$). It is more efficient, and you don't need to implement an inverse routine. This is why XOR is used a lot in cryptography for linear transformations.

0xDEAD10CC - September 10, 2017, 7:27 pm

(Sep 10, 2017 07:19 PM)Phizo Wrote: [\[CITE\]](#) I know very well what your idea was. It is suggested by you prompting its use "as" an encryption algorithm. Not "in." Semantics are very important when conveying ideas. Regardless, I was just saying for everyone, just as you do when you attempt to take someone's post and make it more explicit frequently. I'm not sure why you feel the need to argue, as there is nothing to argue.

Security through obscurity. Not a good principle in general. This has no benefit over traditional bitwise like XOR.

The point was not in regards to my use of "in" or "as" because this could be an encryption algorithm on it's own OR used in combination with other methods. The real point was that you assumed I was guaranteeing any level of adequate security with using such an algorithm by bringing up linear cryptanalysis... An algorithm being broken by such methods do not make any of the existing block ciphers out there undefined for their use in encryption.

I'm just saying, you're arguing level of security, when there was nothing in the context of my post that was related to that for there to be a debate about it.

Phizo - September 10, 2017, 7:29 pm

(Sep 10, 2017 07:27 PM)0xDEAD10CC Wrote: [\[CITE\]](#) The point was not in regards to my use of "in" or "as" because this could be an encryption algorithm on it's own OR used in combination with other methods. The real point was that you assumed I was guaranteeing any level of adequate security with using such an algorithm by bringing up linear cryptanalysis...

Which is precisely the point. Why would you ever suggest an insecure encryption algorithm? What use does an insecure cipher have? It seems you misunderstand the entire point of ciphers. It is you who suggested an adequate level of security by proposing an encryption algorithm. I'm not the one to blame here.

(Sep 10, 2017 07:27 PM)0xDEAD10CC Wrote: [\[CITE\]](#) An algorithm being broken by such methods do not make any of the existing block ciphers out there undefined for their use in encryption.

Of course it does. Broken ciphers (block or otherwise) should never be used. The same way insecure hash algorithms like MD5 should not be used anymore.

(Sep 10, 2017 07:27 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I'm just saying, you're arguing level of security, when there was nothing in the context of my post that was related to that for there to be a debate about it.

Again, it is implicated in the suggestion of the algorithm itself. You're promoting something inherently insecure as something which can/should be used for encryption. There is no good application of your algorithm. It should absolutely not be used in isolation for encryption, and there is no point of using it with an S-box or some similar mechanism to ensure non-linearity, as XOR is much more efficient and has the same benefits and then some.

0xDEAD10CC - September 10, 2017, 7:30 pm

I can understand your point, but I was not recommending it, this is a "(for fun)" thread as indicated by the title when I first posted it many days ago. It seemed like you were trying to make it a point to

me rather than the general public as to the level of insecurity this algorithm could provide. People still learn vingnere's cipher for learning purposes, this could serve as the same as no good encryption method is created out of thin air.

Phizo - September 10, 2017, 7:40 pm

(Sep 10, 2017 07:30 PM)0xDEAD10CC Wrote: [\[CITE\]](#) I can understand your point, but I was not recommending it, this is a "(for fun)" thread as indicated by the title when I first posted it many days ago. It seemed like you were trying to make it a point to me rather than the general public as to the level of insecurity this algorithm could provide. People still learn vingnere's cipher for learning purposes, this could serve as the same as no good encryption method is created out of thin air.

I don't doubt your ability to discern its insecurity. I don't think you're stupid. While I realise this is for fun, your post suggested it can be used for encryption. It really has no good application as a cipher. Of course people learn the Vigenère cipher, both the algorithm itself and how to break it, but it is never implicated that it's applicable today, which your initial post can suggest to people.

POLY MATH - September 10, 2017, 7:50 pm

Guys, I'm just trying to learn something here.

PELock - February 28, 2019, 4:47 am

Pretty pointless code with no real life application. As a cipher is useless.

0xDEAD10CC - February 28, 2019, 6:30 pm

(Feb 28, 2019 05:47 AM)PELock Wrote: [\[CITE\]](#) Pretty pointless code with no real life application. As a cipher is useless.

So are you as a member or you would've read the title of the thread in full. ?

PELock - March 1, 2019, 2:59 pm

You need to learn more about crypto.

0xDEAD10CC - March 1, 2019, 7:51 pm

(Mar 1, 2019 03:59 PM)PELock Wrote: [\[CITE\]](#) You need to learn more about crypto.

The main issue here is your literacy skills. Look at the joke of a thread you have on string encryption.

PELock - March 12, 2019, 1:29 pm

It's worthless as a crypto, don't use it guys, stick to real encryption algorithms.

maryous - March 14, 2019, 10:04 am

idk why people are so negative, the section is slow asf and the dude posted something regardless of what it is and we should thank him instead of saying bs. This code was useful for me since im stepping out of the basic of the language and looking for codes like this to learn things differently and what can the language do .

Thanks mate ?