

Forum:[Hack Forums](#)

Thread: [Basic Encryption Challenge](#)

Anthrax_ - October 25, 2016, 4:16 am

Hey HF -

Recently I've been somewhat dabbling in Encryption and ciphers and such, and I figured hey what the hell might as well make one of my own. This is very basic, but it could potentially be very hard to solve. There are currently two version, basic and challenge. Neither are more difficult than the other per se but the challenge one is much longer and the chances of you guessing what it says halfway through are much lower, BUT, you will probably unlock the entire cipher before solving every word. If you plan to try the challenge I would recommend giving that a stab first. The way to solve the message is in the text and hints, and it's simple. You will most likely need the hints, but it is not impossible to solve without them. All you need is to make connections, and your computer. Hints are in a spoiler in case you don't want to use them, and I can say I will be VERY impressed if you solve and don't use them, but be honest. One hint IS necessary: Any number represents a single digit. I will add solvers if they come in.

I have no doubt in my mind it might take most of you a long time to complete this, if at all. Please comment if you plan on giving this a crack so I know this thread is alive, and it will also be appreciated as this took me like three hours to do, even though the answer to solving it is quite simple and in the text, but that was the hard part. If you are interested in learning how I made my cipher/key, or the key itself please comment or contact me via PM and I will gladly send you it!

The Message

Spoiler ([Click to View](#))

Code 3@)*%211 4!&@!(#!22112 5@!*!%#@#\$12222 3^!%@\$122 5!)@!!#!^(22222
4!%@@@%!*2212 3@)*%211 4!@!@^@%2122 3\$!%&121

HINTS

Spoiler ([Click to View](#)) Every word starts and ends with a number. Every number represents a single digit.(So 576 would be 5-7-6) Figure out what these numbers mean.

*Look down

*The number in front pertains to the word itself, and the letters(but not completely)

*The numbers at the end of the word pertain to the letters. Divide, and I don't mean mathematically

*Do not be mislead by the above hint, the two numbers do have a DIRECT co relation

*Numbers do not represent anything having to do with the actual letters in the words themselves.

*<_1_2_3_4_5_6_7_8_9_10_>

CHALLENGE VERSIONSpoiler ([Click to View](#))

Code 2(@)12 2(!)12 6%!!((%!*112112 2@)!%22 4^(!\$\$1121 3!#%\$212 3@#*!%212
4@#(!@!@2122 9@@@!%!@!@)%%!*22222112 2@)!%22 3\$(%111 4@)*!!\$2112
2@)!%22 4^(!\$\$1121 5@)*!%!(%21221 3@#*!%212 3!!*%121 7@#(!@!@(!&2122121
2@)!%22 6%!\$\$\$@!!*%121221 4!^(!\$2112 4@#(@)*2122 8!^!@)(%!\$#%21211211

Challenge Hints

[1]Spoiler ([Click to View](#))*There are 23 words

[2]Spoiler ([Click to View](#)) The world volunteer = 9 @@!%!@!@)%%!* 22222112

devitup - October 26, 2016, 6:58 am

#1: the quick brown fox jumps over the lazy dog

#2: is it easier to find men who will volunteer to die than to find those who are willing to endure pain with patience

Note: The second phrase has errors. Not sure if this was a transcription problem if you manually typed it, or if it was an issue with what was encoded. The second word should be 2(@)12 and not 2(!)12 to be "it", volunteer is misspelled, and lastly 4@#(@)*2122 ("with") should have been 4@#(@)*2121 instead.

Crypto is a very good field to get into. I'm not sure how much you've studied the subject, but I'd suggest the book "Applied Cryptography: Protocols, Algorithms and Source Code".

Anthrax_ - October 26, 2016, 8:49 am

Wow, thanks for solving! I've posted this on two forums and nobody had even attempted it and I was beginning to become a bit discouraged.

I did manually type it, and I had been working on it for quite some time and I had also pulled an all nighter even before that ;)

If you wouldn't mind, how did you solve it? I mean there's the obvious reasons in the text (do tell which ones though) but was it the hints, or some out of the box method or program?

I'm just interested in how you solved because as I mentioned above, you're the sole solver! :P

devitup - October 26, 2016, 10:44 am

(Oct 26, 2016 08:49 AM) Anthrax_ Wrote: [\[CITE\]](#) Wow, thanks for solving! I've posted this on two forums and nobody had even attempted it and I was beginning to become a bit discouraged.

I did manually type it, and I had been working on it for quite some time and I had also pulled an all nighter even before that ;)

No problem, just thought I would offer up an answer as some encouragement to let you know that it's a very good field of study and it's also good exposure to problem solving. You should definitely use what you learn as a stepping stone to more advanced ciphers and related practices (ex. secure key exchange).

(Oct 26, 2016 08:49 AM) Anthrax_ Wrote: [\[CITE\]](#) If you wouldn't mind, how did you solve it? I mean there's the obvious reasons in the text (do tell which ones though) but was it the hints, or some out of the box method or program?

I didn't use any special program, just about an hour and some pen/paper. I looked at the character frequency of the whole set and found it was 10 values that were not numbers, the numbers at the end of each block added up to the total length of those sets and were never higher than 2, etc. Looking at the blocks with a length of 2, there were only 2 unique values for the left-most character (! and @). Writing those out and seeing all values less than 27 spelled out the rest.

If you haven't taken up Python, I'd say give that a go and see if you can implement an encoder and decoder there. It's actually very approachable and slices make working with string groups like this very simple. Then work on implementing Vigenère and Caesar ciphers. I used to have an MSDOS program that would let you interactively attack substitution ciphers like this, but it isn't the one on the Internet Archive and I'd need to dig out some floppies to even see if that one made it out of the BBS world.

Your post took me back to people posting giant blobs of hex on the cryptanalysis usenet groups. They'd obviously be blasted constantly because they would almost universally be using one-time pads with no key information and claim "I made something unbreakable!". Algorithms go under open, public review for a reason. Still, processing power increases in the last 20+ years have made things very interesting and revised a lot of perspectives for acceptable key length and made unapparent issues apparent. It's never a question of if something protected by encryption is crackable, but rather for how long it would take. While some places don't take too well to "break my encryption" challenges, don't stop improving yourself.

Thanks for the thread, btw.

Anthrax_ - October 26, 2016, 10:32 pm

(Oct 26, 2016 10:44 AM)devitup Wrote: [\[CITE\]](#) No problem, just thought I would offer up an answer as some encouragement to let you know that it's a very good field of study and it's also good exposure to problem solving. You should definitely use what you learn as a stepping stone to more advanced ciphers and related practices (ex. secure key exchange).

I didn't use any special program, just about an hour and some pen/paper. I looked at the character frequency of the whole set and found it was 10 values that were not numbers, the numbers at the end of each block added up to the total length of those sets and were never higher than 2, etc. Looking at the blocks with a length of 2, there were only 2 unique values for the left-most character (! and @). Writing those out and seeing all values less than 27 spelled out the rest.

If you haven't taken up Python, I'd say give that a go and see if you can implement an encoder and decoder there. It's actually very approachable and slices make working with string groups like this very simple. Then work on implementing Vigenère and Caesar ciphers. I used to have an MSDOS program that would let you interactively attack substitution ciphers like this, but it isn't the one on the Internet Archive and I'd need to dig out some floppies to even see if that one made it out of the BBS world.

Your post took me back to people posting giant blobs of hex on the cryptanalysis usenet groups. They'd obviously be blasted constantly because they would almost universally be using one-time pads with no key information and claim "I made something unbreakable!". Algorithms go under open, public review for a reason. Still, processing power increases in the last 20+ years have made things very interesting and revised a lot of perspectives for acceptable key length and made unapparent issues apparent. It's never a question of if something protected by encryption is crackable, but rather for how long it would take. While some places don't take too well to "break my encryption" challenges, don't stop improving yourself.

Thanks for the thread, btw.

Well, like you said: problem solving.

Hackers are, quite literally, by definition, problem solvers. I figured this would do very well here, and I see it did! Thanks really, this gives kind of a good feeling knowing somebody sat down for an hour trying to solve a problem I wrote :)