

Re: How good an encryption algorithm is this?

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2004-11/5361.html>

From: Bonj (*Bonj_at_discussions.microsoft.com*)

Date: 11/24/04

Date: Wed, 24 Nov 2004 03:11:03 -0800

> *But surely there are enough CryptoAPI examples to help you out, aren't there?*

Not that work, no.

> *True, but I suspect most people who *were* trained in encryption would have understood the reasons given for not trying to come up with your own algorithm to start with.*

You're contradicting yourself...

What about the people that *have* invented the *good* encryption algorithms? Were they trained in encryption? If so, then by your principle "they should have known not to bother trying to invent their own", but if they weren't trained, then how did they manage to invent a good algorithm? What I take out of this is that, obviously, *either*, people are wrong in that home grown algs are not *necessarily* worse (albeit certainly no better) than professional ones, OR, you don't actually need to be particularly clever or trained in cryptography to invent a professional crypto algorithm, just by fluke – happen to stumble on some tricky principle or something. I suspect it's the latter though...

> *That's really not a good argument in favour of creating your own algorithm. Just because someone hasn't already studied yours in an attempt to crack it doesn't mean it's more secure than one which many highly skilled people *have* studied and failed to find significant problems with. It just means they haven't looked at yours yet.*

Mmmm, maybe so. It's not really a good argument in favour of creating my own, but since a better description of my problem perhaps would be "how to persist the key on the client in secrecy", then it's not all that strong an argument against it either.

>
> > *but haven't looked at yours and are*
> > *unlikely to".*
> >

microsoft.public.dotnet.languages.csharp: Re: How good an encryption algorithm is this?

> > *Again, weak as it is, the point comes back – if they're not likely to look*
> > *at it, how do they crack it?*
>
> *They crack it when it's worth cracking, rather than now. Do you really*
> *want to only find out whether your algorithm is strong enough when it's*
> *already deployed in the field, and is protecting real–world data?*
>

Again , I would say that I've posted a link to a google groups thread which is someone else having exactly the same problem, but perhaps described in a better way than I am capable of (and certainly hammered out quite a lot)!

> *But why did you wait until you'd made what you could see was a bad*
> *choice before posting, rather than just asking for help with the*
> *CryptoAPI in the first place?*

Because I feared people would gloss over the post. At least if I post my own attempt,

a) I can't be vindicated hitting a brick wall and not trying any further
b) If people thought it was good, then hopefully they would point it out.
c) If people thought my algorithm is bad in itself or a bad idea, they'll be more keen to point out how to do the CryptoAPI in order to set me on the right track.

> *It's like starting to design your own*
> *string class because you can't get Substring working – you're far more*
> *likely to get help on standard stuff than you are to get genuinely*
> *expert opinions on your own custom algorithms.*
>

> > *Igor has very kindly addressed that, and at the end of the day that's*
> > *probably what I'll go with, but since I started out on this tack and*
> > *spent all of a whole hour writing the damn thing, I'll finish.*
>

> *No–one can stop you, of course – but I'd urge you not to make the*
> *mistake of deciding to actually *use* it at any point, just because*
> *you'll then have it.*
>

> > > *In what way is that simply repeating the assertion?*
> >

> > *Well, no – in light of the above paragraph, it possibly just seemed*
> > *like that – but you have explained yourself sufficiently now and I*
> > *thank you for that.*
>

> *Goodo – I'm glad we understood each other in the end.*
>

> --

> *Jon Skeet – <skeet@pobox.com>*

> *<http://www.pobox.com/~skeet>*

> *If replying to the group, please do not mail me too*
>

Re: How good an encryption algorithm is this?