

Re: Easy test of surrogate factoring

Source: <http://sci.tech-archive.net/Archive/sci.math/2005-02/6948.html>

From: Matt Gutting (*tchrmatt_at_yahoo.com*)

Date: 02/18/05

Date: Fri, 18 Feb 2005 09:56:21 -0500

Matt Gutting wrote:

> *jstevh@msn.com* wrote:

>

>> *Paul Rubin* wrote:

>>

>>> *Here's an even easier test of surrogate factoring: I gave you a list*

>>> *of fifty 20-digit composite numbers a couple weeks ago and you*

>>

>>

>> *haven't*

>>

>>> *factored a single one. I think I know what test score to assign,*

>>> *based on that result.*

>>

>>

>>

>> *No, it's not a fair test. I'll explain why, again, and I'll also*

>> *explain again, what the test is, and why my test is an easy one.*

>>

> *[snip]*

>

>> *If it's not random, then some way exists to figure out how to get all*

>> *the integer Az 's without knowing M .*

>>

>> *Those are the two possibilities.*

>>

>

> *That's true, James. And if you look and listen, I believe you'll find that*

> *everyone here agrees that the second possibility is the case.*

>

>> *Now posters working hard to discount this research--apparently deciding*

>> *that it's Harris mathematics or JSH algebra--are avoiding the actual*

>> *issue from what I've seen in looking at posts in this thread.*

>>

>> *The reason is simple, the mathematics doesn't work for them either way,*

>> *if their intent is to discount it.*

>>

>> *That's what happens with major math results, human beings working hard*

sci.math: Re: Easy test of surrogate factoring

>> *to dismiss them have to ignore facts.*
>>
>> *Now my test is simple: calculate Ax, by taking an M with known factors,*
>> *and *look* at the prime factors of the denominator of Ax, and you will*
>> *find that they are the same prime factors as T has.*
>>
>> *That proves that there are rules governing the value of the rationals*
>> *Ax's that results from the integer Az's.*
>>
>> *And then, algorithms *can* be developed.*
>>
>> *Now, if mathematicians were what they claimed then it wouldn't be*
>> *required that I actually produce a working program that can factor RSA*
>> *numbers, as that's an unreasonable request. It's an unreaonable*
>> *request as if I were at that point I wouldn't need to convince anyone,*
>> *as I could just demonstrate.*
>>
>> *Worse, my ability or lack of ability to write such a program does not*
>> *disprove the mathematics!*
>>
>> *And in this case that means someone else might, or may already have.*
>>
>> *But mathematicians are NOT what they claim, and in this case they are*
>> *setting other people up to be responsible for their failures, if things*
>> *go badly.*
>>
>> *Now I'll work at building that program to factor an RSA number, but*
>> *while time passes, it's you who may suffer the consequences, in a world*
>> *where hackers may now have the upper hand.*
>
>
> *Heck, James, *I* can write a program to factor an RSA number. How about*
> *this?*
>
> *int M;*
> *M = //insert the RSA number here*
> *for (int i=2; i<=sqrt(M);i++) {*
> *if (M % i = 0) {*
> *cout << i << " is a factor of" << M;*
> *}*
> *}*
>
> *This will produce *every* factor of M less than sqrt(M) (and the only*
> *reason*
> *it doesn't stop after just one factor is because I don't know enough C++).*
>
> *Of course, it will take quite a while. This is precisely what others*
> *have been*
> *saying: not that your math can't produce a workable algorithm, but that*
> *it's*
> *(at the very best) no more efficient than the least efficient algorithms*

sci.math: Re: Easy test of surrogate factoring

- > *currently existing. I think it's great that you've (apparently) arrived at*
- > *a new algorithm, but it's not important if it's not an improvement over*
- > *what's already available.*
- >
- > *Matt*
- >
- > *(excess newsgroup stripped)*

Oops, sorry. Apparently some have been saying that your math can't (always) produce a workable algorithm. And you know what? After looking at some of their examples, I have to agree with them.

Matt