

Re: ANNOUNCE: New "Leopard6" CSPRNG !

Source: <http://www.derkeiler.com/Newsgroups/sci.crypt/2003-02/3013.html>

From: Bob Jenkins (bob_jenkins@burtleburtle.net)

Date: 02/19/03

From: bob_jenkins@burtleburtle.net (Bob Jenkins)

Date: 19 Feb 2003 13:22:59 -0800

Paul Crowley <paul@JUNKCATCHER.ciphergoth.org> wrote in message news:<87of59xwf9.fsf@saltationism.subnet.hedonism.cluefactory.org.uk>...

> mrsjunecarey@aol.com (Mrsjunecarey) writes:

> > All the information you need including free source code is available from:

> >

> > <http://wizardsworks.org/~robin/leopard.html>

>

> Note that based on her record here, this is almost certainly

> insecure. Avoid.

It *is* insecure. I sent him a description of how to break it.

Unfortunately, hadn't read my mail for a few days so he posted it first. I'll try to be better in the future. There are some algorithms near what he's doing that I can't break off the top of my head, so there is some hope of something interesting here.

The guts of his generator are

```
#define SIZE 256
int i;
unsigned char x, y, s[SIZE];
for (i=0; i<SIZE; ++i) s[i] = i;
while (TRUE) {
  ++y;
  for (i=0; i<SIZE; ++i) {
    report(s[(s[x]+s[y])%SIZE]);
    temp=s[x]; s[x]=s[y]; s[y]=temp;
    ++x; ++y;
  }
}
```

which is admirably short and amenable to analysis.

So, analysis. For the 256 consecutive results with $x=y$, nothing gets swapped, and it always reports $s[2*s[x]]$. Spotting 256 consecutive results with no more than 128 distinct values should be pretty easy, that tells you where $x=y$. For the next 255 results with $x+1=y$, $s[\text{old}_y]$ is swapped with $s[\text{new}_y]$, so $\text{state}[y]$ is constant for all those results, so it reports $s[s[x]+\text{constant}]$, that is all the values

sci.crypt: Re: ANNOUNCE: New "Leopard6" CSPRNG !

of $s[]$ in some (so far unknown) order without repetition (skipping $s[y]$). Also statistically easy to spot. If you know (or guess) $s[y]$, you know where to stick $s[y]$ in that list, and it gives you a permutation of $0..255$. The contents of $s[]$ at any point of time is also a permutation of $0..255$. The internal state is $p(i)$ and the results are $p(p(i))$. If $s[y]=0$, those two permutations will have the same set of cycles, except any cycles of even length in the internal state will become two cycles of half the original length. It's easy to tell what the cycles of a permutation are, and it's easy to rotate them until you find a $p(i)$ that produces the $p(p(i))$ that is the results. Voila, the internal state. For even cycles sometimes there are multiple possibilities of what the internal state was, but I imagine examining a few more results should narrow it down to the one true answer.