64-bit numbers in perl

Joe Pruitt, 2005-06-06

A question recently came up in the iControl forums about how to deal with the statistics returned by our iControl interfaces. In 9.0, we changed the interfaces to return true 64-bit values consisting of a structure of a low and high 32-bit value (this is due to the fact that there is a lack of support in the SOAP encoding specs, and some of the languages, for native 64-bit numbers).

Our SDK samples show something like this:

```
$low = $stat->{"low"};
$high = $stat->{"high"};
$value64 = ($high<<32)|$low;
```

The question was asked, how come the value is truncated to a 32-bit number? I asked myself that same question as it worked for my build. After some digging, it turns out that perl needs to be compiled specifically to allow include support for 64-bit numbers. You can check your installation out by using the **Config** module

```
use Config;
($Config{use64bitint} eq 'define' || $Config{longsize} >= 8) &&
    print "Supports 64-bit numbers\n";
```

So, what do you do if you don't have control to rebuild your version of perl? Well, thanks to a user who pointed me this way, there exists a perl module to support large numbers. As long as you have the **Math::BigInt** package installed, you can use it as follows:

```
use Math::BigInt;
$low = $stat->{"low"};
$high = $stat->{"high"};
$value64 = Math::BigInt->new($high)->blsft(32)->bxor($low);
```

Happy Coding!

-Joe