****
Two-Way SHA (A collision-resistant file hash sum)
****

This method will likely make sha256 an unbreakable
algorithm for files and whatever else, who knows if
this method is useful for x509 certificates and such?

The concept is simple. I understand that all I ever
need for my hash table concepts is Bob Jenkin's
famous one-at-a-time hash. Hashing the key one way
gives one result, and when you reverse all the bits
of the key and hash again, expect a different result.

This allows me to do the type of stuff I need to do,
with, say, a linear hash for use on database tech.

To do this with sha256 is easy. You need an implementation
of that algorithm, where someone has the correctly
coded algorithm for you.

Take that code, which is MSB-LSB.

Do one hash sum that way. Do the second hash sum
LSb-MSb, which is the reverse of the file in bits,
NOT BYTES. With all bits reversed, if the sha256
algorithm does its job, we have an entirely different
sha hash sum LSbit-MSbit.

Then we need to know the file length.

This then forms our combined file hash sum, which is
hash(MSB-LSB), hash(LSbit-MSbit), filelen.

That should make collisions nearly impossible, for
some proper sha-ish hash algo. That said, good luck
for some hacker to break it and insert a malicious file.

That's the generalized idea, and I'm not sure if the
community is aware of it.

-0/1. The 1.