From: Regenscheid, Andrew (Fed)
To: Bassham, Lawrence E. (Fed)
Subject: SUPERCOP architecture
Date: Thursday, May 4, 2017 8:30:42 PM

I spent a couple hours looking at the SUPERCOP source code, but for the life of me I can't figure out what's going on with even simple things. I'm hoping you can help figure out the basic stuff.

There's a handful of things I think I know:

1) SUPERCOP can handle at least C, C++ and Assembly code. Maybe more.
2) Implementations sometimes call external libraries. OpenSSL is used quite a bit.
3) Some libraries are included in SUPERCOP. Notably, GMP.
4) Many libraries aren't included. Notably, OpenSSL.

Also, I see a lot of #includes on header files that don't exist. For example, I see a lot of files have: **#include crypto_hash_sha256.h**

In those cases, they call SHA-2 with **crypto_hash_sha256(…)**

So, presumably they're pulling in a SHA-2 implementation from somewhere. But where? There's no **crypto_hash_sha256.h** file. And, as far as I can tell, that's not in some sort of common shared library that could reasonably be expected on most machines. So what's going on here?

Doing some searching, it looks like **crypto_hash_sha256.h** probably comes from either NaCL or its fork, libsodium. But how does that get there?

Did you ever try running SUPERCOP on a machine at NIST? Does it download other libraries before it runs? Or does it fail trying to run a lot of algorithms if you don't have things like libsodium installed?

-Andy