Transition to SHA-256 including Alternative Means

Update Nov. 2014

Communication Plan for Moving to SHA-2 Signed Assertions, approved by InCommon Steering in Oct. 2014

Update June 2014 - Approval of Alternative Means

As of June 2014, InCommon Steering approved an Alternative Means to help manage the transition to SHA-256. To read the full document click here. For the original problem statement click here. At the time of approval, Steering asked the AAC to return in August 2014 with a communication plan to help the assurance community move forward in supporting this new NIST/FICAM requirement.

Transition to SHA-2

Background

In sp800-131A, published in January 2011, NIST updated its recommendations for use of several cryptographic algorithms and key lengths for use in the US Federal government. In particular, it calls for discontinuing use of the SHA-1 digest function or hash algorithm in digital signatures effective January 1, 2014 and recommends using any of the digest functions known collectively as SHA-2 for use in digital signatures.

Although NIST recommendations need not be binding outside of the US Government, InCommon must pay attention for two reasons. The first is that the InCommon Identity Assurance program references NIST recommendations with regard to cryptographic algorithms. Basically, crypto algorithms used in a Bronze or Silver compliant implementation either must be approved by NIST or be covered by an InCommon-approved Alternative Means (AM). So campuses wanting Silver or Bronze must either use SHA-2 or InCommon must approve an alternative they can use. So unless an AM would say "it's ok to continue using SHA-1 for digital signatures", some change to widely deployed behavior is needed, and for that appropriate testing and validation needs to be done to ascertain how it will work and how to advise deployers accordingly.

So why shouldn't an AM be approved that says "SHA-1 is still ok"? Recall that an approved AM must demonstrate that the alternative means addresses the risk addressed by the means it is alternative to in a comparable or superior manner. So the only possible basis for approval is if switching to SHA-2 would introduce greater risk to operations than sticking with SHA-1. That could be the case if a sufficient number of production SPs with which a given IdP interoperates are not capable of processing SHA-2 signatures from the IdP, or are not capable of processing assertions signed variously by different IdPs using SHA-1 or SHA-2 signatures. Hence, adequate testing needs to be done that will either validate that almost all SPs play well with SHA-1 and SHA-2, or that many do not and fixing them is non-trivial.

The second reason InCommon should pay attention to sp800-131A is that, as a matter of best practice, NIST has moved the bar up a notch. InCommon should implement best practices to continue to earn the trust of its members; that is its essential reason for being. InCommon should enable participants wishing to achieve Bronze or Silver to achieve best practice.

What needs to be done?

The following proposals are offered to start discussion. I'm confident that the collective experience of TAC members will improve upon these.

I'll note that there is reason to be optimistic. Searching for a little while indicates that releases of most SAML implementations in the last year or two that are prevalent in InCommon support SHA-2. I'm less certain how well some of these (non-Shibboleth) releases handle a mix of SHA-1 and SHA-2.

Prepare a Shibboleth IdP extension that makes it easy to switch to SHA-2

For the Shibboleth IdP, an extension to change the algorithm the IdP uses to sign SAML messages already exists in a rudimentary form. A further enhancement is needed that would enable an IdP operator to deploy the extension in a simple fashion. Future shibboleth IdP releases should incorporate a simple means to select the digest function to be used in signing assertions.

The doc at the URL above also contains some detailed information about prerequisites and caveats for enabling SHA-2 for IdPs or SPs.

Test handling of SHA-1 and SHA-2 by InCommon SPs

Arrange to send an unsolicited SHA-1 signed response or assertion and a SHA-2 signed response or assertion to each production SP in InCommon and see how they respond. The results should help us identify a set of SPs that accept SHA-1 but not SHA-2. Probably should contact a subset of SP operators to learn about releases, configurations, and operating environments that produce the problem, and assess prospects for remedying.

Even though all current SPs handle SHA-1, sending both SHA-1 and SHA-2 would help us determine when a failure of the test procedure is unrelated to the digest algorithm.

Depending on the results of this testing, a next step would either be focused on disseminating documentation to deployers about how to enable SHA-2 in their operation, or if the data indicate that shifting to SHA-2 would expose many InCommon members to unacceptable operational risk, an Alternative Means would be written on that basis.

Cf. a more detailed page on testing.

Who should do what?

The SHA-2 Shibboleth IdP extension would best be one that exists in a single, established code namespace, so that each deployment does not need to deal with namespace issues. It would also be simplest if the resulting extension and corresponding wiki doc exist on the shibboleth.net site in suitable locations. Perhaps this is best done by the Shibboleth developers.

A TAC subcommittee should quickly be chartered and spun up to do the testing. Liaison with the InCommon Assurance Advisory Group would be desirable so that they can monitor and see whatever the result will be at the earliest opportunity. We should aim for the testing to be done and analysis and follow ups complete by end of August 2013. A fun summer project for a couple of inspired techies! Also, we can ask technical leads affiliated with common SAML implementations to help us.