From:	Chang, Shu-jen H. (Fed)
To:	Dang, Quynh H. (Fed); Sonmez Turan, Meltem (Fed); Perlner, Ray A. (Fed); Chang, Shu-jen H. (Fed); Kelsey, John M. (Fed); internal-hash
Cc:	(b) (6)
Subject:	Re: Current KMAC document in Word
Date:	Wednesday, April 27, 2016 3:17:26 PM

Thank you all for your comments. I think I got more than I bargained for — so many comments to address. :-) I thought I'd just help John to clean up his draft…..

Anyway, I'm focusing on reviewing the PQC CFS now, so will address your comments after my review is done.

Shu-jen

From: Quynh Dang <quynh.dang@nist.gov>
Date: Wednesday, April 27, 2016 at 2:22 PM
To: "Sonmez Turan, Meltem (Assoc)" <meltem.turan@nist.gov>, Ray Perlner <ray.perlner@nist.gov>, "Chang, Shu-jen H. (Fed)" <shu-jen.chang@nist.gov>, John Kelsey <john.kelsey@nist.gov>, "internal-hash@nist.gov" <internal-hash@nist.gov>
Cc: "(b) (6)
Subject: Re: Current KMAC document in Word

Attached are my comments.

One big comment is about output length encodings.

Sponge construction was designed in a way to improve efficiency: having two phases: absorbing and squeezing so that additional output bits can be generated very efficiently without the need to recompute (rehash) the whole message again and without using output length encodings. And, this construction has been adopted by the community and there have been no problems about its security as far as I know.

Now, with output length encodings, the efficiency is destroyed.

As explained before, there are no needs for output length encodings.

1) I have not seen any protocols/applications where output bits from a symmetric function (or a deterministic function) are reused (in the case of the same message, but different signatures are needed for different recipients, the data blocks to be signed contains context information to distinguish themselves beside using different signing keys). I also don't see any situation where the output bits need to be reused so that the related outputs property becomes a problem.

2) There are no output length encodings for HMAC with 96 or 64-bit outputs which has the related output property with the full-output HMAC. There have been no security issues with this approach because output bits from a HMAC are not reused.

3) There are no requirements preventing truncating SHA-512 and SHA3-512 to get 448-bit hash values. I hope we are not going to create a new mode on top of each of these functions to add output length encodings to avoid the related outputs property.

4) If someone wants to make us look bad, he or she pretends to specify a protocol where the output bits are reused. To deal with this situation, we just need to make it clear that output bits shall not be reused.

Also, as currently written, if somebody does not want to use output length encodings, he or she must use the default cSHAKE: (1) to build a different keyed mode (such as just prepending a fixed size key to input messages) and (2) as a hash function etc... because KMAC, TupleHash and FPH require output length encodings. Those new derived functions from the default cSHAKE can't use the L and S formats and must use some implicit domain-separation strings as parts of an input message, when domain-separation strings are desired.

Without output length encodings, everything will go very smooth.

Regards,
Quynh.

From: Sonmez Turan, Meltem (Assoc) <meltem.turan@NIST.GOV>
Sent: Tuesday, April 26, 2016 4:02:53 PM
To: Perlner, Ray (Fed); Chang, Shu-jen H. (Fed); Kelsey, John M. (Fed); internal-hash
Cc: (b) (6)
Subject: RE: Current KMAC document in Word

Shu-jen and John,

I attached my comments. I was almost done with my comments, when Ray sent his; so I could not merge them.

A few highlights

- <!--[if !supportLists]--><!--[endif]-->Our notation is not consistent with FIPS 202 (explained in the attached file.)
- <!--[if !supportLists]--><!--[endif]-->We need a security discussion for tuple hash. It is not clear what we aim for in the SP.
- <!--[if !supportLists]--><!--[endif]-->Max_integer needs to be specified.
- <!--[if !supportLists]--><!--[endif]-->Etc.

Thanks,
Meltem

From: Perlner, Ray (Fed) [mailto:ray.perlner@nist.gov]
Sent: Tuesday, April 26, 2016 3:32 PM

To: Chang, Shu-jen H. (Fed) <shu-jen.chang@nist.gov>; Kelsey, John M. (Fed) <john.kelsey@nist.gov>; internal-hash <internal-hash@nist.gov>
Cc: (b) (6)
Subject: RE: Current KMAC document in Word

Here are my comments.

From: Chang, Shu-jen H. (Fed) [mailto:shu-jen.chang@nist.gov]
Sent: Monday, April 25, 2016 2:21 PM
To: Kelsey, John M. (Fed) <john.kelsey@nist.gov>; internal-hash <internal-hash@nist.gov>
Cc: (b) (6)
Subject: Re: Current KMAC document in Word

Folks,

Attached is the current draft that combines the two SPs. I think it's ready for review, even though I have not gone back to check Keccak team's comments to see if I have missed addressing any of those. I also think it's possible to "trim the fat" (i.e., remove the repetitions) further. Please give me your comments by COB 5/2/2016 (next Monday).

John,

Please note that I also changed the title of this SP; let me know if you object to any of the major changes.

Thanks,

Shu-jen

From: "Chang, Shu-jen H. (Fed)" <shu-jen.chang@nist.gov>
Date: Wednesday, April 20, 2016 at 6:23 PM
To: "Sonmez Turan, Meltem (Assoc)" <meltem.turan@nist.gov>, John Kelsey <john.kelsey@nist.gov>, "internal-hash@nist.gov" <internal-hash@nist.gov>
Cc: "(b) (6)
Subject: Re: Current KMAC document in Word

Folks,

I agreed with Meltem, so I decided to try to merge the two documents into one. Attached is what I have come up with so far. This is still work in progress, so please don't bother to review it yet. I'm sharing the draft just for you to see how it's organized now.

I have also talked to John, and he didn't have any objection about this effort.

Meltem,

I've moved Appendix A to Sec. 2.3, which I think is what you wanted; I also like this organization better.

Shu-jen

From: "Sonmez Turan, Meltem (Assoc)" <meltem.turan@nist.gov>
Date: Tuesday, April 12, 2016 at 5:24 PM
To: John Kelsey <john.kelsey@nist.gov>, "internal-hash@nist.gov" <internal-hash@nist.gov>
Subject: RE: Current KMAC document in Word

Hi everyone,

I attached my comments on the KMAC, TupleHash, FPH draft.

Summary:

<!--[if !supportLists]--><!--[endif]-->Looking at this draft, I see that having a separate SP for Cshake did not really simplify this specification of the functions. We still need the description of all intermediate functions such as string encode(), bytepad, right_enc(), left_enc() and entire Appendix A in both SPs.

<!--[if !supportLists]-->- <!--[endif]-->Appendix B can be merged into the document. Or can be restructured to get rid of repetition of the pseudocodes.

<!--[if !supportLists]-->- <!--[endif]-->Restrictions on the input variables are not clear, e.g., in Tuplehash we require S to be a byte string, but we don't specify a similar restriction for other functions, or max_integer variable used in FPH is not defined. We always check the validity of L in the pseudo code, but never do it for the length of S.

<!--[if !supportLists]-->- <!--[endif]-->Pseudocodes can be simplified, many unnecessary temporary variables.

<!--[if !supportLists]-->- <!--[endif]-->We require the length of the subblocks to be a power of 2, in Tuplehash. Why is this necessary ?

Meltem

From: Kelsey, John M. (Fed) [mailto:john.kelsey@nist.gov]
Sent: Tuesday, April 12, 2016 2:08 PM
To: internal-hash <internal-hash@nist.gov>
Subject: Current KMAC document in Word

Everyone,
This is the version of the document I sent out on the hash forum. If anyone wants to make comments, I'm interested. I tried to address everyone's comments from before, but I'm sure I missed some. Any comments are useful, but big-picture stuff is more valuable at this point than typos.
--John