SHA-256 Today and Maybe Something Else in a Few Years: Effects on Research and Design

Panelists: Niels Ferguson (Microsoft)
Antoine Joux (Univ Versailles-S-Q-e-Y)
Bart Preneel (Kath Univ Leuven)
Ron Rivest (MIT)
Adi Shamir (Weizmann)

Moderators: Paul Hoffman (VPN Consortium)
Arjen Lenstra (EPFL)

Overview

- Given today’s reality (SHA-1 is widely deployed and has collision-resistance problems, SHA-256 exists and is getting wide deployment), what should the crypto community do?
- After this hour, we have a discussion slot, so please hold your comments until then

Four topics

- What problems with SHA-1 and SHA-256 do we really face?
- What properties of hash functions do we know we need for the long term?
- Should we develop one all-purpose hash or several special-purpose hash functions?
- How do we design the next algorithm(s)?

What problems with SHA-1 and SHA-256 do we really face?

- Given the attacks on the collision-resistance of SHA-1 and the close relationship between the designs of SHA-1 and SHA-256, how much confidence do you have in the collision-resistance of SHA-256?
- Does collision-resistance really matter? How much lower than 2^{63} attack-effort would really worry you?
- Do you have any worries about pre-image resistance of SHA-1 or SHA-256 ?

What properties of hash functions do we know we need for the long term?

- Are there features other than collision-resistance and pre-image-resistance that are important for hash designs? If so, what are they and how do we rank them?
- What “practical” features (for instance, speed and one-pass processing) are mandatory and which are just useful?
- If one considers collision weaknesses relatively unimportant but pre-image weaknesses disastrous, are we better off with creating new Merkle-Damgård functions or going to different designs?

Develop one all-purpose or several special-purpose hash functions?

- Can we describe the properties of specialized hashes well enough so that implementers will get it right?
- Does the community need multiple lengths of output? If not, what should the single hash-length be?
- What is currently the strongest design for non-Merkle-Damgård hashes?

How do we design the next algorithm(s)?

- Do we need to encourage more novel attacks on SHA-family to better understand hash function design, or is the SHA-approach a dead end?
- Given the prevalence of SHA-256 today, how do we encourage researchers to start designing replacements sooner rather than later?
- Do we need to start afresh or do we just need more tweaks?
- How does the U.S. NSA patent over parts of the design of SHA-256 (US patent 6829355) affect the development of new hash functions?