

SHA-3 update

Quynh Dang
Computer Security Division
ITL, NIST

SHA-3 Competition

11/2/2007

SHA-3 Competition Began.

10/2/2012

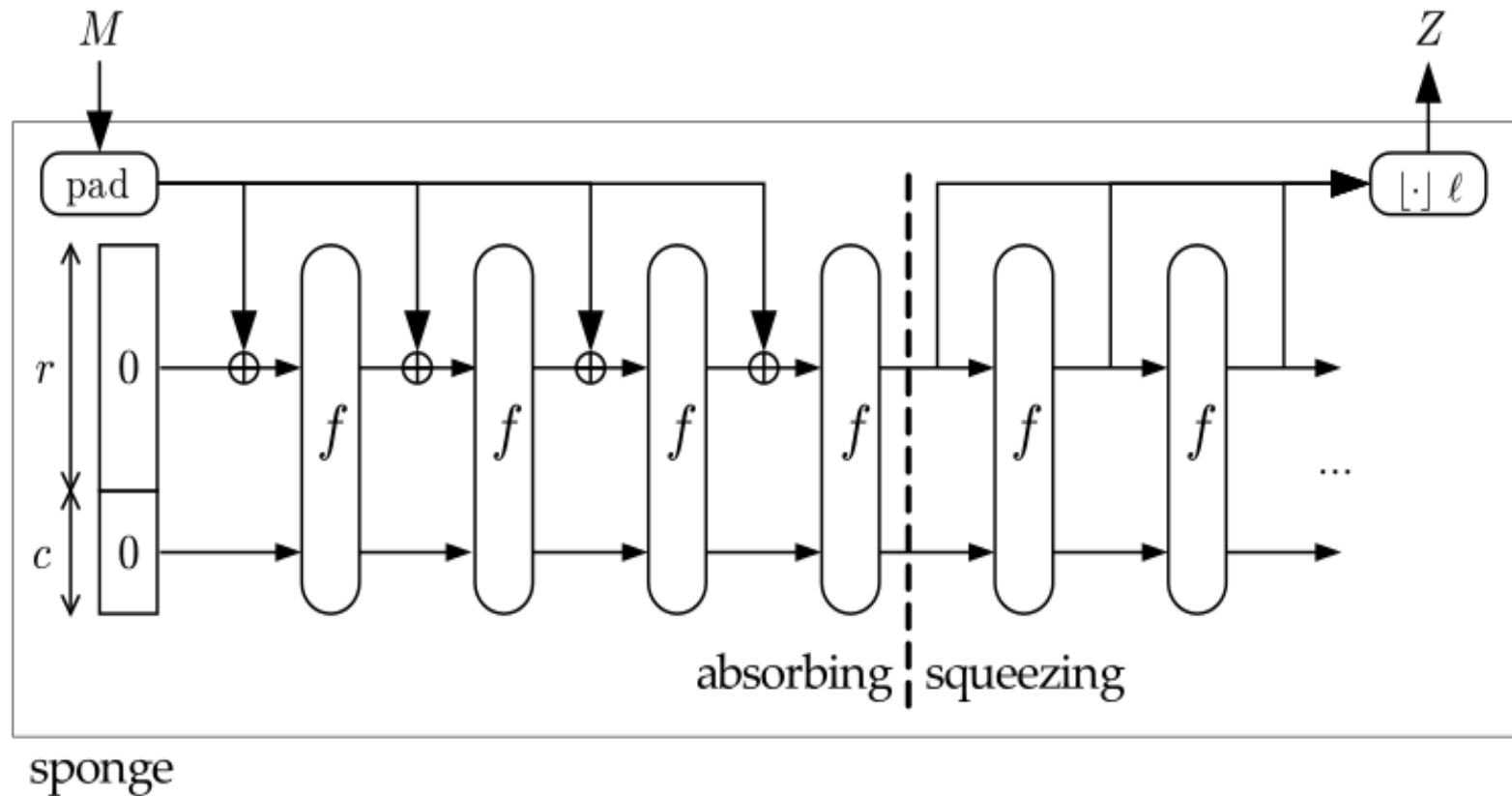
Keccak announced as the SHA-3 winner.

Secure Hash Algorithms Outlook

- ▶ SHA-2 looks strong.
- ▶ We expect Keccak (SHA-3) to co-exist with SHA-2.
- ▶ Keccak *complements* SHA-2 in many ways. Keccak is good in different environments.

Keccak is a sponge - a different design concept from SHA-2.

Sponge Construction



Sponge capacity corresponds to a security level: $s = c/2$.

SHA-3 Selection

- ▶ We chose Keccak as the winner because of many different reasons and below are some of them:
 - ▶ It has a high security margin.
 - ▶ It received good amount of high-quality analyses.
 - ▶ It has excellent hardware performance.
 - ▶ It has good overall performance.
 - ▶ It is very different from SHA-2.
 - ▶ It provides a lot of flexibility.

Keccak Features

- ▶ Keccak supports the same hash-output sizes as SHA-2 (i.e., SHA-224, -256, -384, -512).
- ▶ Keccak works fine with existing applications, such as DRBGs, KDFs, HMAC and digital signatures.
- ▶ Keccak offers flexibility in performance/security tradeoffs.
- ▶ Keccak supports tree hashing.
- ▶ Keccak supports variable-length output.

Under Consideration for SHA-3

- ▶ Support for variable-length hashes
- ▶ Considering options:
 - ▶ One capacity: $c = 512$, with output size encoding,
 - ▶ Two capacities: $c = 256$ and $c = 512$, with output size encoding, or
 - ▶ Four capacities: $c = 224$, $c = 256$, $c = 384$, and $c = 512$ without output size encoding (preferred by the Keccak team).
- ▶ Input format for SHA-3 hash function(s) will contain a padding scheme to support tree hashing in the future.
- ▶ NIST will standardize 224, 256, 384 and 512 alternative hashes to the 4 hash sizes of SHA-2.

Other Features for standardization considerations

- ▶ NIST will look into the possibility of standardizing another authenticated encryption scheme using Keccak permutation (the Duplex mode) in the future.
- ▶ NIST will also look into the possibility of using smaller permutations of Keccak for lightweight applications in the future!

Comments

NIST's Crypto Toolkit:

<http://csrc.nist.gov/groups/ST/toolkit/index.html>.

Thanks to the security area directors for this presentation opportunity!

Any comments/questions?

Discussion mailing list: Hash-forum@nist.gov

Comments for NIST: internal-hash@nist.gov