KangarooTwelve
draft-viguier-kangarootwelve-02

Benoît Viguier[1]

CFRG Meeting, July 17, 2018

[1] Radboud University, Nijmegen, The Netherlands

Yet another hash function?

No RFC exists with a hash function that . . .

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size

Yet another hash function?

No RFC exists with a hash function that ...

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis
  - 35 third-party cryptanalysis papers in 10 years of KECCAK/SHA-3 cryptanalysis (`https://keccak.team/third_party.html`)

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis
  - 35 third-party cryptanalysis papers in 10 years of KECCAK/SHA-3 cryptanalysis (`https://keccak.team/third_party.html`)
  - more cryptanalysis than SHA-256 and/or SHA-512 (we counted about 21)

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis
  - 35 third-party cryptanalysis papers in 10 years of KECCAK/SHA-3 cryptanalysis (`https://keccak.team/third_party.html`)
  - more cryptanalysis than SHA-256 and/or SHA-512 (we counted about 21)
  - For reduced-round KECCAK, best attacks seem to stabilize to
    - 5 rounds for collision and (second) preimage attacks

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis
  - 35 third-party cryptanalysis papers in 10 years of KECCAK/SHA-3 cryptanalysis (`https://keccak.team/third_party.html`)
  - more cryptanalysis than SHA-256 and/or SHA-512 (we counted about 21)
  - For reduced-round KECCAK, best attacks seem to stabilize to
    - 5 rounds for collision and (second) preimage attacks
    - 8 rounds for distinguishers

Yet another hash function?

No RFC exists with a hash function that . . .

- supports arbitrary output length: XOF rather than a hash function
- provides **scalable parallelism** increasing with input size
- is based on a permutation that won the open worldwide SHA-3 competition
  - reuse of code and/or hardware for FIPS 202, (e.g. in ARMv8.2 instruction)
  - inherently **faster** than FIPS 202 and SP-800-185
- is a public design **and** has vast amount of 3^{rd} party cryptanalysis
  - 35 third-party cryptanalysis papers in 10 years of KECCAK/SHA-3 cryptanalysis (`https://keccak.team/third_party.html`)
  - more cryptanalysis than SHA-256 and/or SHA-512 (we counted about 21)
  - For reduced-round KECCAK, best attacks seem to stabilize to
    - 5 rounds for collision and (second) preimage attacks
    - 8 rounds for distinguishers

    KANGAROOTWELVE has 12 rounds.

Why is it interesting for the IETF?

- KECCAK/KANGAROOTWELVE is an open design
  - Public design rationale
  - Result of an open international competition
  - Long-standing active scrutiny from the crypto community
- Best security/speed trade-off
  - Speed-up w/o wasting cryptanalysis resources
  - Proven generic security (sponges, tree)
- Scalable parallelism
  - As much parallelism as the implementation can exploit
  - Without additinal parameter