NIST's views on SHA-3's security requirements and Evaluation of attacks

National Institute of Standards and Technology

Presented by

Mridul Nandi

Outline of the talk

- Security requirements
- Cost of an attack
- Categorization of attacks
  - Completely broken
  - Wounded
  - Undermining confidence
  - Little to no concern
- Summary

Quick View on Security Definitions

Collision attack: find $M \neq M'$ s.t. $H(M) = H(M')$

Preimage attack : given h, find M s.t. $H(M) = h$

2nd Preimage attack : given M, find $M' \neq M$ s.t. $H(M) = H(M')$

Length-ext^n: Given H(M) and |M|, find h' and z s.t. $H(M \,||\, z) = h'$

eTCR: Find M and then for a randomly chosen r, find M' and r' s.t. $H_r(M) = H_{r'}(M')$, $(r,M) \neq (r',M')$ where H_r is the randomized hash

PRF-attack: Distinguish $HMAC_K$ based on H from a random function

Some Best Known Generic Attacks

- Collision (parallel collision search): $O(2^{n/2})$ computation, $O(1)$ memory (for each processor), $2^{n/4}$ parallelization
- Preimage (trial and error attack): $O(2^n)$ comp^n, $O(1)$ memory, $2^{n/2}$ parallel^n
  - time-memory trade-off ignores off-line computation
- Distinguishing HMAC based on collision (for MD-type hash algorithms)

Some Best Known Generic Attacks contd.

- eTCR, 2nd preimage, no such generic attack less than 2^n computation
  - some generic attacks on MD or with other similar structures
    - Kelsey-Schneier attack
- MD has length extension attack

Comparing with generic attacks

- Any attack requiring more computations than generic attacks, can be ignored.
- Beating generic attack w.r.t. both time and memory also beats w.r.t. computation, but converse may not be true.
- Is there reason to ignore attack better than generic attack w.r.t. computation?

Security Requirements of SHA-3

An n-bit Hash Algorithm expects roughly

1. n/2-bit Collision and PRF-security (HMAC)
2. n-bit preimage and length extension security
3. (n-k)-bit 2nd preimage (the target message has length 2^k) and enhanced Target Collision security (for randomized hash only)

m-bit truncation expects at least the above securities with m replacing n

Other Security Considerations

- Multi-collision attack (more than two messages with same collision value)
  - narrow-pipe designs are vulnerable
- More than (n-k)-bit security for 2nd preimage with length shorter than 2^k
  - ideally n-bit security
- Resistance against these attacks is viewed positively

Q and A

1. measuring n-bit security
2. significance of having security beyond n/2-bit security (e.g., preimage)

Evaluation of attacks

Cost of an attack

- Computational (time) complexity
  - off-line and online computation
  - in sequential attack, computation = time
  - in parallel attack, computation $\leq$ time x # processor
  - success probability of an attack is related to computation

- Memory complexity and Parallelizability
  - parallelizability, memory, etc. are important factors for attack's performance considerations

Cost of an attack

- How do we compare two attacks where one requires more time while the other requires more memory?
  - one-dimensional metric
  - we assume there exists parallel version of attacks (unless strong evidence provided against it)

Broad Categorization Of Attacks

1. Completely breaks (practical threat)

2. Wounds (fail to satisfy security requirements)

3. Undermines confidence (some weakness)

4. Little to no concern

Undermines Confidence

- Variants of attacks: near collision, pseudo collision, low margin reduced round attack, etc.
- Reduced round attacks limit performance
- Unexpected properties of hash or its components
  - nonrandom behaviors (failing statistical test).
  - block ciphers: not random permutations
  - weakness in S-box

Undermines Confidence

- Flawed understanding of designers
  - flawed proofs or assumptions, demonstrating a property that was "proved" in submission not to exist

- Many attacks (or maybe observations) will not violate collision or preimage resistance
  - still probably care about these

Undermines Confidence

- Rule of thumb:
  - we never care about attacks at > 2^n work
  - observations are worrying if they get substantially below the theoretical limit
  - no hard and fast rule measuring an observation

- How do we evaluate observations?

Wounded Hash Algorithms

- Any attacks with computation less than its corresponding expected complexity (NIST's requirement)
- Beating 2^n computation bound preimage
  - Ex: n=256, computation = memory = 2^{128}
    - (if sequential) parallel generic attack with same time and memory exists but not w.r.t. computation
- Similarly for other security requirements

If it’s broken, we’re done with it

- Attack with both $comp^n$ and memory below the numbers
- Computation is based on collision bound with some buffer.

Hash Size	log_2(Comp)	log_2(Mem)
224	100	80
256	120	100
384	180	150
512	240	200

If it's broken, we're done with it

- 1 bit memory ≈ 2^{20} to 2^{40} hash computations (an estimate based on current technology, should be subject to periodic review)

- Analogy with AES and DSA key sizes

Hash Size	$\log_2$(Comp)	$\log_2$(Mem)
224	100	80
256	120	100
384	180	150
512	240	200

Summary

- At this point, we evaluate attacks based on how they will affect our choice in the next round
- That means asking, for any given attack:
  - Does it completely break the hash function?
  - Does it violate NIST security requirements?
  - Does it undermine our confidence in the hash function?
  - Does it require the hash to be unacceptably slow to resist the attack?

Q and A

- How do we measure n-bit security?
- Significance for having beyond n/2-bit security? (e.g., preimage)
- How do we compare two attacks?
- How do we evaluate an observation?
- Others?

Comments?