Current status of MD5 and SHA-1

Eric Rescorla
Network Resonance
ekr@networkresonance.com

Review of hash function terminology

Collision Find M, M' st $H(M) = H(M')$

1st preimage Given X, find M st $H(M) = X$

2nd preimage Given M, find M' st $H(M') = H(M)$

In a perfect hash function of length l:

- Collisions require $2^{l/2}$ effort to find
- 1st and 2nd preimages require 2^l effort to find

The current situation

MD5 Collisions can be easily found [details to appear in Eurocrypt 2005]

SHA-1 Collisions in SHA-1 with 2^{69} effort (design goal = 80 bits) [this just out on Feb 15]

- ... in theory. Too expensive to find an actual collision

Certificates Lenstra et al. demonstrate a pair of certificates with different public keys but the same hash (and hence signature) [Feb 29!]

Important limitations:

- None of these attacks allows you to compute a preimage
- The colliders are not totally controllable
- Which pair collides depends on current hash state

Implications of this attack

DON'T PANIC!

- Not affected
  - Key derivation functions (PRFs)
  - Peer authentication without non-repudiation (SSL, IPsec, SSH, etc.)
  - Message authentication (HMAC)
  - Challenge-response protocols (probably)
- Affected
  - Non-repudiation (at least technically)
  - Certificate issuance — but only in some special cases
  - Timestamps (maybe)

The Lenstra certificate attack (approximately)

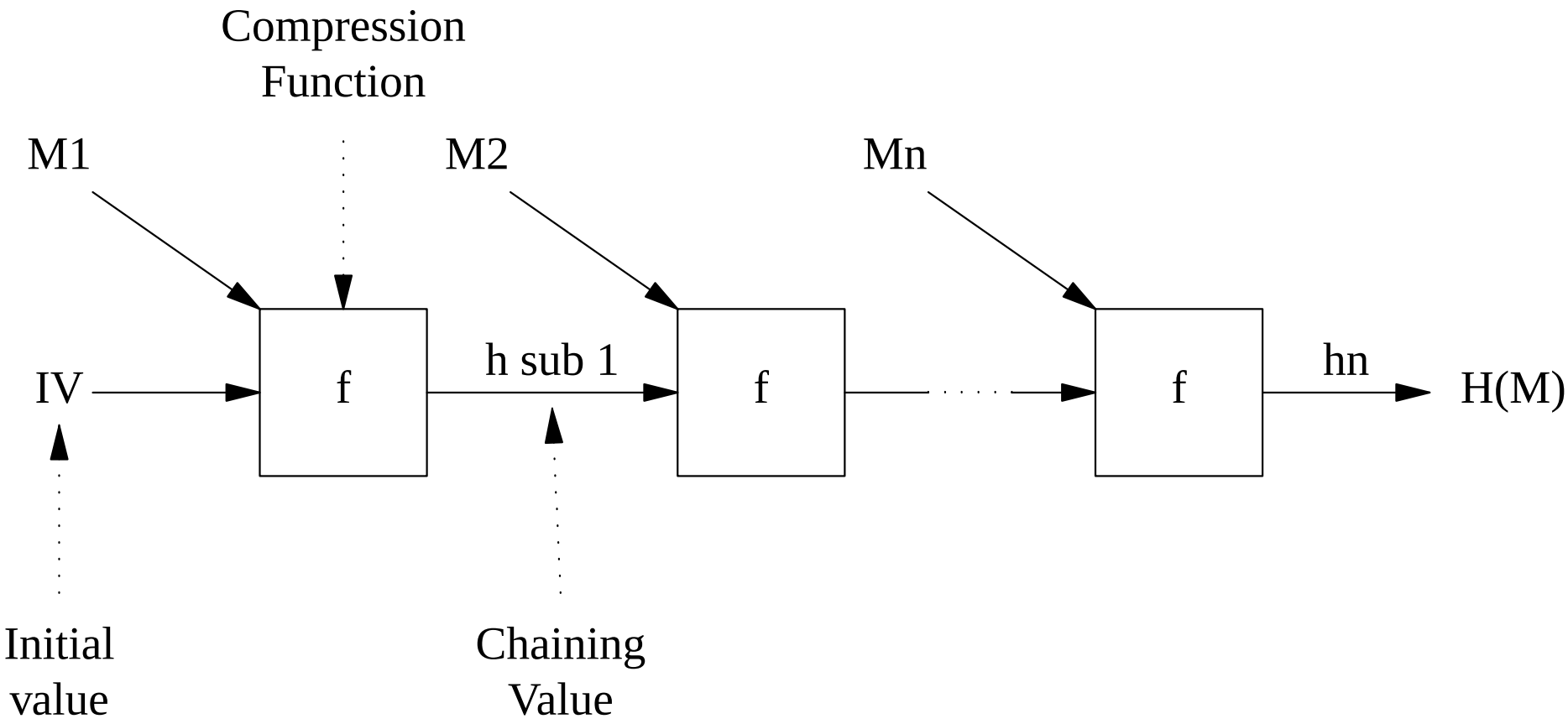
- Start with a certificate template T
  - version, serial, signature algorithm, issuer, validity, subject
- And a pair of colliding 512-bit values A and B
- Find a value X such that $A||X$ and $B||X$ are valid RSA public keys.
- Get a cert signed over $A||X$
  - This is also a cert with $B||X$
- This only works when you know T
  - Which means predicting serial and validity
  - Not necessarily possible with a real CA
- Extensible to name collisions? Maybe, but not controllable yet.

Moving forward

- New hash functions
  - SHA-224 and greater
    - Probably more secure than SHA-1—but we're not sure
    - Trivial protocol changes—specify new OIDs
  - Something entirely new
    - Probably a block-cipher-based hash like Whirlpool, MDC-2, MDC-4
    - Requires writing new documents (paging NIST...)
- Randomized hash algorithms
  - Transmit $Random, Sign(H(Random||MSG))$
  - Requires some protocol changes (in AlgId in ASN.1)
- Randomize cert serial numbers (or dates)
  - Only blocks attacks on cert issuance
  - Backward-compatible change to CA procedure

Supplementary material

Merkle-Damgard Construction



After [Shrimpton 2004]

Why is HMAC OK?

- $HMAC(key, M) = H((key \oplus opad)||H((key \oplus ipad)||M))$
- Recovering *key* means a preimage attack—and may not be information theoretically possible, especially with a truncated HMAC
- Forging also means a preimage attack
- Generating a colliding pair requires knowing the state
  - Which is key dependent and therefore secret
  - HMAC security proof depends on random state collision-freeness, not generic collision-freeness

Do we know enough to select a new hash function?

- All MD4-based functions are now questionable
- We don't have a good theory of hash construction
- Best available candidates are based on block ciphers
  - There's a provability gap
    - 12 constructions are provably secure in ideal cipher model [Shrimpton]
    - But not in the PRP model [Simon]
  - No rate one construction is secure
- Answer: No!