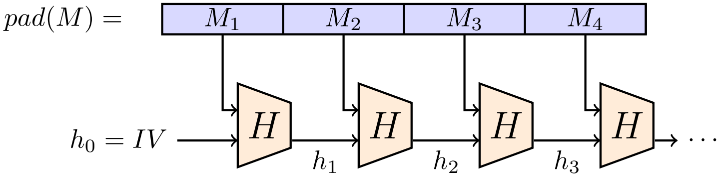
Cryptanalysis
Part II: Cryptanalysis of Hash Constructions

André Schrottenloher

Inria Rennes
Team CAPSULE

Merkle-Dåmgard

Let $H : \underbrace{\{0,1\}^n}_{\text{Chaining value}} \times \underbrace{\{0,1\}^m}_{\text{Message block}} \to \{0,1\}^n$



Fact

If H is collision-resistant, and pad is an appropriate padding scheme, $\mathcal{H} = MD[H]$ is collision-resistant.

Preliminaries

Collisions

From a given chaining value h, find two blocks x, x' such that $H(h, x) = H(h, x')$: $\mathcal{O}(2^{n/2})$.

Preimage

From a given chaining value h and target t, find a block x such that $H(h, x) = t$: $\mathcal{O}(2^n)$.

Multi-target preimage

From a given chaining value h and set of targets T, $|T| = 2^t$, find a block x such that $H(h, x) \in T$: $\mathcal{O}(2^{n-t})$.

$\Longrightarrow$ all of this assumes nothing of the function H.

Length Extension on Merkle-Dåmgard

Length extension attack

Attack

Given $\mathcal{H}(x)$, where x is unknown, obtain $\mathcal{H}(x \| \mathrm{pad}(x) \| y)$ for arbitrary suffix y.

Length extension attack

Attack

Given $\mathcal{H}(x)$, where x is unknown, obtain $\mathcal{H}(x \| \mathrm{pad}(x) \| y)$ for arbitrary suffix y.

- We know the final state after absorbing $x \| \mathrm{pad}(x)$
- Restart from this state and compute the next chaining values ourselves (incl. padding)

Avoiding this

Solution

Use a different compression function for the last call.

Second Preimage on Merkle-Dåmgard

Second preimage attack

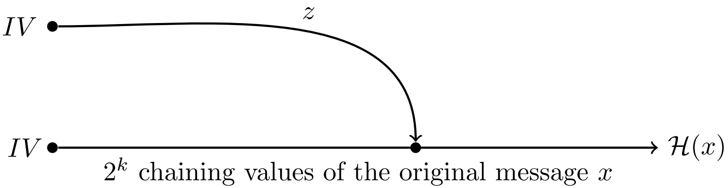
Consider a very long message $x = x_0 \| x_1 \ldots \| x_{2^k - 1}$, with 2^k chaining values.

Objective

Given $x, \mathcal{H}(x)$, find $y \neq x$ such that $\mathcal{H}(y) = \mathcal{H}(x)$.

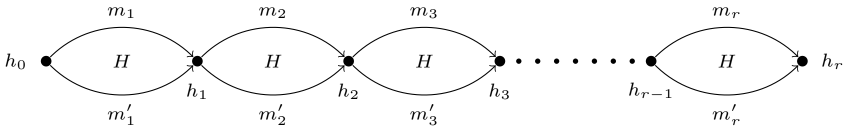
If the padding did not **depend on the message length**, this would be easy:

- Find z such that $\mathcal{H}(z)$ falls on a chaining value (time $\mathcal{O}(2^{n-k})$)
- Concatenate z with the rest of the message



Problem: the two messages have different lengths.

Interlude: multicollisions in MD



- Start from a chaining value h_0
- Find a collision from h_0: let h_1 be the output
- Find a collision from h_1: let h_2 be the output
- ...

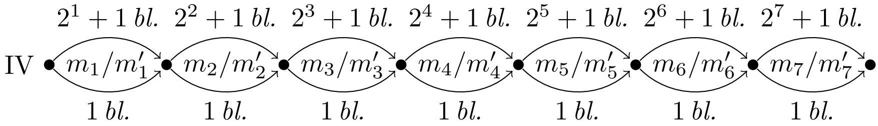
Every choice of message $(m_1 \text{ or } m'_1)\|(m_2 \text{ or } m'_2)\|\ldots\|(m_r \text{ or } m'_r)$ leads to the **same value** h_r.

We can compute a 2^r-collision in time $\mathcal{O}\left(r2^{n/2}\right)$.

How much space do we need to store it?

Expandable message

- So far all the messages in the multicollision have the same length.
- New idea: use messages of different block lengths.



- First collision: 1 block vs. $2^1 + 1$ block
- Second collision: 1 block vs. $2^2 + 1$ block
- ...

Theorem

For any $r \leq j < r + 2^r$, we can produce a message (by choosing m_i or m_i' blocks) with output h_r and length i **blocks**. The structure is constructed in time $\widetilde{\mathcal{O}}(2^r + 2^{n/2})$.

$\Longrightarrow$ multicollision with length control.

Time to construct the EM structure

Naively: we need r collisions, the last one between a message of 2^r blocks and a message of 1 block.

$\implies \mathcal{O}\left(2^{r+n/2}\right)$ complexity

Time to construct the EM structure

Naively: we need r collisions, the last one between a message of 2^r blocks and a message of 1 block.

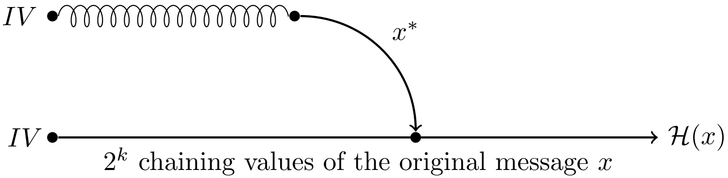
$\implies \mathcal{O}\left(2^{r+n/2}\right)$ complexity

Cleverly

- For each collision of 1 block vs. $2^i + 1$ block, we fix the 2^i first block to dummy values.
- Now the total amount of compression function calls is:

$$1 + \ldots + 2^r + \mathcal{O}\left(r2^{n/2}\right) = \widetilde{\mathcal{O}}\left(2^r + 2^{n/2}\right)$$

Second preimage attack (ctd.)



1. construct a 2^k-expandable message: $\widetilde{\mathcal{O}}\left(2^k + 2^{n/2}\right)$ with output h_k
2. find x^* such that $H(h_k, x^*)$ is one of the chaining values: $\mathcal{O}\left(2^{n-k}\right)$
3. select in the EM the message having the right length

- Total: $\mathcal{O}\left(2^k + 2^{n/2}\right) + \mathcal{O}\left(2^{n-k}\right)$
- Corresponding message has 2^k blocks (optimal for $k = n/2$, but long message)

Avoiding this

Solution

- Increase the internal state (**wide-pipe** construction): instead of n bits, have $2n$ bits
- At the end, compress the $2n$ bits into n bits (typically: truncate)

Nostradamus Attack

Nostradamus attack scenario

Nostradamus says: "I can predict the lottery output".

- Nostradamus publishes a hash output h
- After the lottery outputs x, Nostradamus shows that $h = \mathcal{H}(x \| s)$ where s is an arbitrary (garbage) suffix

Nostradamus concludes: "I have correctly predicted x".

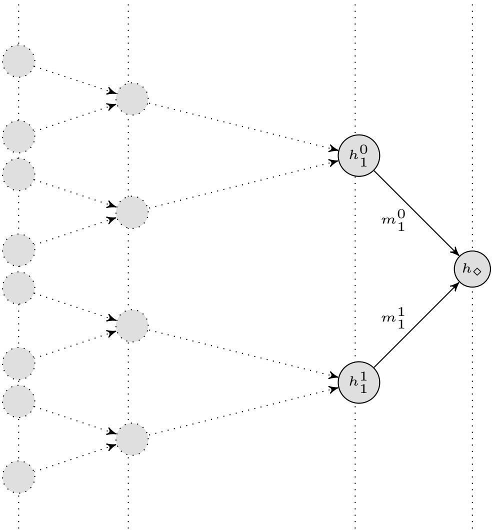
Chosen target forced prefix pre-image resistance:

Given x and h, find s such that $h = \mathcal{H}(x \| s)$.

For Merkle-Dåmgard, CTFP is **easier** than preimage.

The diamond structure

Find many messages leading to the same hash value.



The diamond structure (ctd.)

1. Start from 2^k random chaining values.
2. Find message pairs which map the 2^k chaining values to 2^{k-1} (many collisions)
3. Find message pairs to map the 2^{k-1} values to 2^{k-2}
4. ...

Naive complexity: $\mathcal{O}\left(2^k \times 2^{n/2}\right)$.

The diamond structure (ctd.)

1. Start from 2^k random chaining values.
2. Find message pairs which map the 2^k chaining values to 2^{k-1} (many collisions)
3. Find message pairs to map the 2^{k-1} values to 2^{k-2}
4. ...

Naive complexity: $\mathcal{O}\left(2^k \times 2^{n/2}\right)$.

Better complexity:

- At each level, select $2^{n/2+k/2}$ extensions ($2^{n/2-k/2}$ per current value).
- Expect $(2^{n/2+k/2})^2 2^{-n} = 2^k$ collisions (enough to form all collision pairs).

Result: $\widetilde{\mathcal{O}}\left(2^{k/2+n/2}\right)$.

The herding attack

1. Nostradamus creates a diamond structure, publishes the output h
2. On challenge x, Nostradamus finds a message m such that $h(x, m)$ is in the first level of the diamond

Complexity: $2^{n/2+k/2} + 2^{n-k}$, balanced with $k = n/3 \implies \mathcal{O}\left(2^{2n/3}\right)$.

Conclusion

- All of these attacks are **generic**: they are limitations from the constructions, not the primitives.
- Basic Merkle-Dåmgard has many hurdles: exercise caution
- Modern hash functions (SHA-3) are more often built using **Sponges** than MD