Cryptography, a walk into SHA1

Kevin Grandemange

November 28, 2013

What does a hash function do ?

- It maps data of variable length to a digest of fixed length

$$h : D \rightarrow \{0,1\}^n$$

- A hash is supposed to represent its data only
- Has a legal value. Used in digital signature

Examples of use of hash function

- password verification
- compare by hash
- virus protection
- integrity
- non-repudiation

Property of a good hash function

A hash function must be:

- **One-way**
  Given $h \in \{0,1\}^n$, finding M such that H(M) = h should not be significantly faster than hashing 2^n random messages with H.
- **Second-Preimage Resistant**
  Given $M \in \{0,1\}*$ finding M' such that H(M) = H(M') should no be significantly faster thant hashing 2^n random messages with H.
- **Collision-Resistant**
  Finding two distinct messages M and M' such that H(M) = H(M') should not be significantly faster thant evaluating H about $2^{n/2}$ times

Construction of a hash function

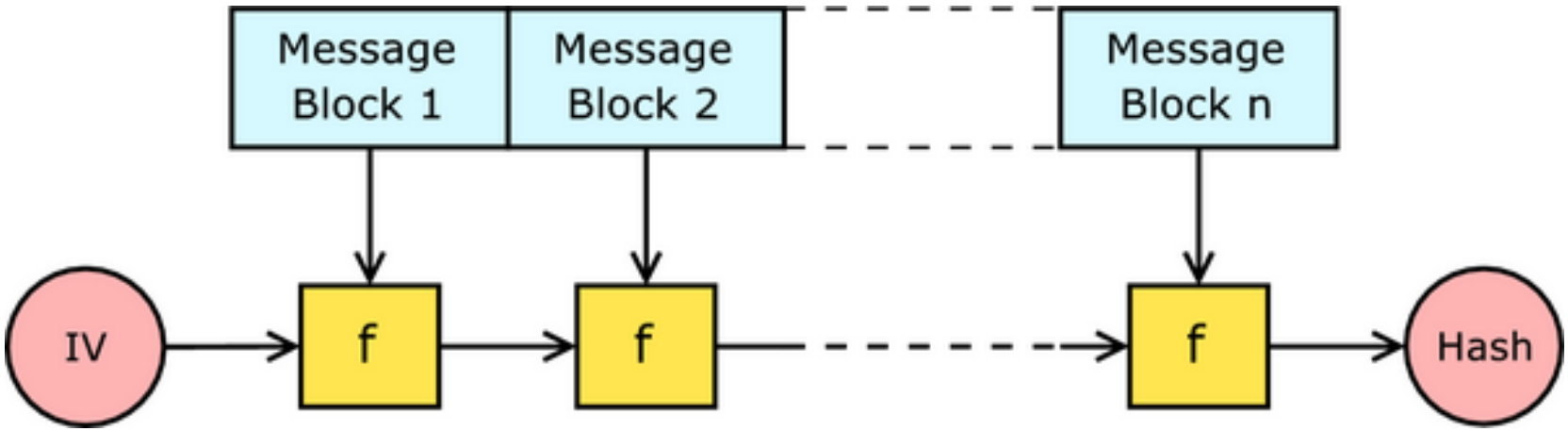
A hash function before Sha3 is the product of:

- *a compression function*
  hash a small part of the message into a smaller part
- *a mode of operation*
  describe how the compression function is used for bigger message

In most of the hash function, the Merkle-Damgard construction is used

Merkle-Damgard construction

IV: Initial Value, f: compression function



How does Sha1 work ?

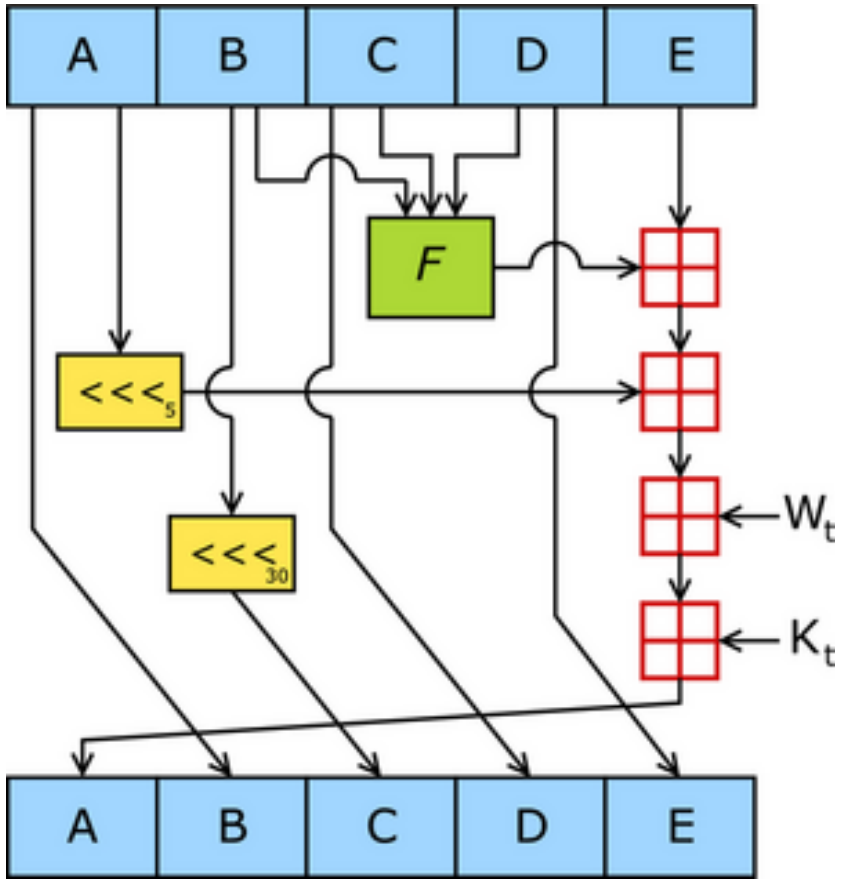
- The digest is 160 bits long
- initial message cut into blocks of 512 bits
- Each block is processed in the compression function

$$H : \{0,1\}^{512} x \{0,1\}^{160} \rightarrow \{0,1\}^{160}$$

- current block is cut in 16 words of 32 bits W_t
- W_t are then expanded to 80 words according to the following formula:

$$W_i = W_{i-16} \oplus W_{i-14} \oplus W_{i-8} \oplus W_{i-3} \lll 1, 16 \leq i \leq 79 \quad (1)$$

Compression function: Example of Sha1



The concatanation of the 32 bits words A,B,C,D,E is the intermediate hash K_t are constants of the algorithm This function is called for each of the 80 W_t with the previously calculated A,B,C,D,E

Equation of Sha1

$$PAS_{i+1} : \begin{cases} A_{i+1} = (A_i \lll 5 + f_i(B_i, C_i, D_i) + E_i + K_i + W_i \\ B_{i+1} = A_i \\ C_{i+1} = B_i \ggg 2 \\ D_{i+1} = C_i \\ E_{i+1} = D_i \end{cases} \quad (2)$$

Details of Sha1

Boolean functions and constants used in SHA-1

pas i	$f_i(B, C, D)$	K_i
$0 \leq i \leq 19$	$f_{IF} = (B \wedge C) \oplus (\bar{B} \wedge D)$	$0x5a827999$
$20 \leq i \leq 39$	$f_{XOR} = B \oplus C \oplus D$	0x6ed6eba1
$40 \leq i \leq 59$	$f_{MAJ} = (B \wedge C) \oplus (B \wedge D) \oplus (C \wedge D)$	0x8fabbcdc
$60 \leq i \leq 79$	$f_{XOR} = B \oplus C \oplus D$	0xca62c1d6

Last Step

the last step is called Feed-Forward: The sum modulo 2^{32}: $(A_0 + A_{80})$ $(B_0 + B_{80})$ $(C_0 + C_{80})$ $(D_0 + D_{80})$ $(E_0 + E_{80})$ are concatenated in order to form the result of the compression function: H_t

Birthday Attack

- Algorithm:
  for i=1,..,q do $x_i = D; y_i = h(x_i)$
  if $\exists i, j$ ($i \neq j$ and $y_i = y_j$ and $x_i \neq x_j$) then return x_i, x_j else return FAIL
- Complexity : $2^{n/2}$. For Sha1 : 2^{80}
- Consequences: Not computable, we need to find another method

Differential Attack

Find a set of differences to apply to a message M in order to obtain another message M' wich will satisfy with a good probability H(M) = H(M')

very difficult because of the avalanche effect

Solution: construction of a linear model of Sha1

Linear Model of Sha1

- First approximation: We don't care about the message expansion
- Second approximation: The compression function becomes

$$PAS_{i+1} : \begin{cases} A_{i+1} = A_i \lll 5 \oplus B_i \oplus C_i \oplus D_i \oplus E_i \oplus K_i \oplus W_i \\ B_{i+1} = A_i \\ C_{i+1} = B_i \ggg 2 \\ D_{i+1} = C_i \\ E_{i+1} = D_i \end{cases} \quad (3)$$

Local Collision

- What is the effect of the difference I introduce and how can I suppress it ?
- A local collision is the introduction of one difference and then of a set of differences which will eliminate the effect of the first difference

Notation for the differential path

Table: A reduced set of all the notations used in [2] in order to represent the differents states possibles for a couple of bits x and x'

(x,x')	(0,0)	(1,0)	(0,1)	(1,1)
?	√	√	√	√
-	√	-	-	√
x	-	√	√	-
0	√	-	-	-
u	-	√	-	-
n	-	-	√	-

Representation of a local collision

Table: differential path leading to a local collision

pas	differential path of W_i
i	----------u-
i+1	----------n------
i+2	----------n-
i+3	x----------
i+4	x----------
i+5	x----------
i+6	----------

Using local collision for a global collision

- Because of the expansion of the message, the bits we used in order to create a local collision perturb the result later
- Disturbance vector:

$$V_i : \begin{cases} (2^j) \text{if we initiate a local collision at step i on the } j^{th} \text{ bit} \\ 0 \text{ else} \end{cases} \quad (4)$$

- the disturbance vector describe where the local collisions start.
- To find a real collision, the disturbance vector must follow the same recurrence formula as the expansion of the message.

Back to the real Sha1

- Welcome back to a world of pain and suffering ...
- What was certain in our linear world become probability in the real Sha1.
- There is a probability of $1/2$ that the flipping of our bit didn't destroy everything we worked so hard to create...
- A local collision has a probability of 2^{-5} of success

Demonstration: Finding a Collision on an altered sha1

Bonus: Example of the use of a collision (md5)