Authentication

- It is a mechanism to verify the integrity of the transmitted message or person initiating it.

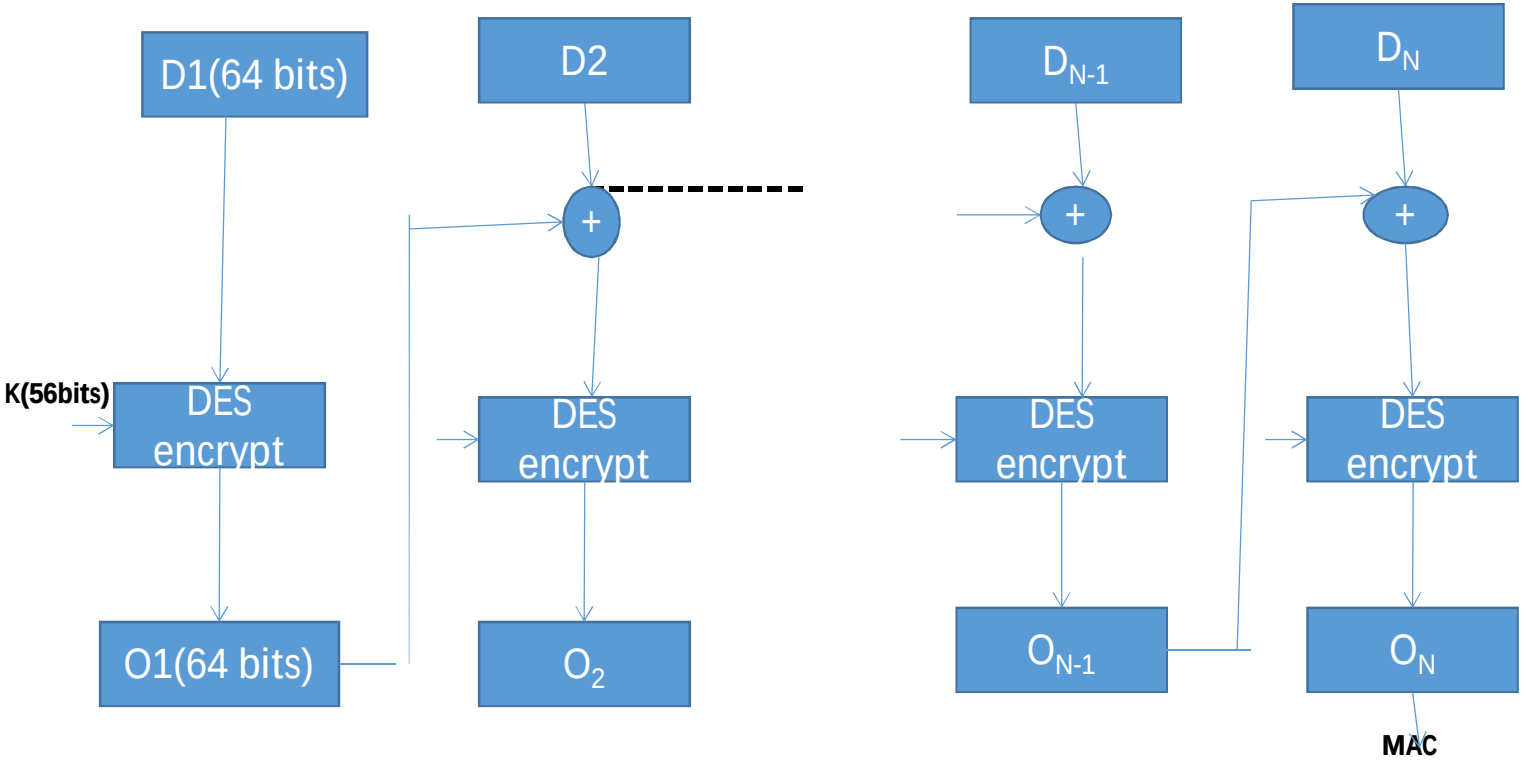
-Entity authentication

- Data authentication

- Message encryption
- Message Authentication Code(MAC)
- Hash functions

MAC

- Data Authentication Algorithm:



Using Hash for Authentication

- A computes Hash value using hash algorithm and concatenates with original data and send to B
- B computes Hash value using same algorithm on the received original data and compares the computed hash value with received hash value
- If both are same then message is authenticated.

Hashing

Properties of Hash function:

1.H can be applied to a block of data of any size

2.H produces a fixed length output

3.H(M) is relatively easy to compute for any given M

4. For nay given value h,it is computationally infeasible to find M such that H(M)=h. This is sometimes referred as one way function **(Preimage resistant)**

5.For any given message M,it is computationally iintractable to find $M^1 \neq M$ such that $H(M^1)=H(M)$::: **weak collision resistance(2nd Preimage resistant)**

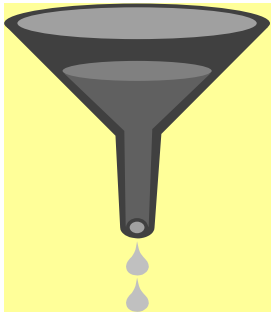
6.It is computationally iintractable to find any pair (M, M^1) such that $H(M)=H(M^1)$::**strong collision resistance(Collision resistance)**

Requirements for Hash Functions

1. can be applied to any sized message M
2. produces fixed-length output h
3. is easy to compute h=H(M) for any message M
4. given h is infeasible to find x s.t. H(x)=h
   - *one-way property*
5. given x is infeasible to find y s.t. H(y)=H(x)
   - *weak collision resistance*
6. is infeasible to find any x,y s.t. H(y)=H(x)
   - *strong collision resistance*

MD5: Message Digest Version 5

Input message(Varaible size)



Output 128 bits(Fixed)

How MD5 works?

- Step 1: Padding
- Step2 : Append length
- Step3 :Devide the input into 512-bit blocks
- Step4: Intialize chaining varaibles

A	Hex	01	25	45	67
B	Hex	89	AB	CD	EF
C	Hex	FE	DC	BA	98
D	Hex	76	54	32	10

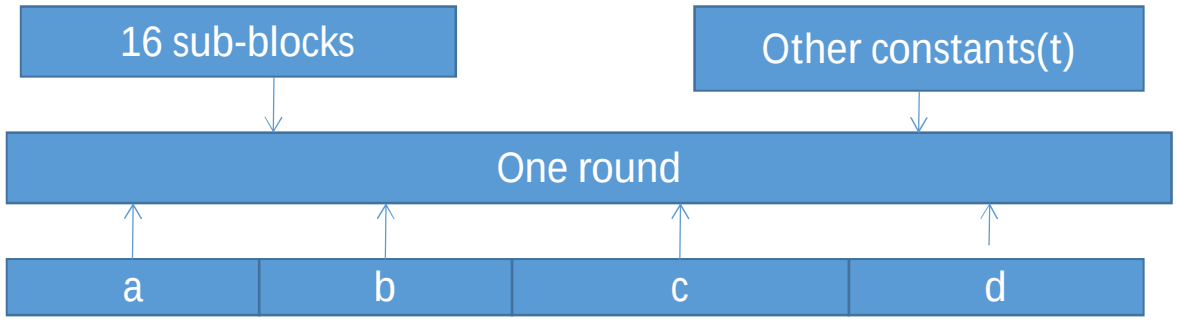
- Step 5: Process blocks

5.1: copy the chaining varaibles into four correspobding varaibles a,b,c and d

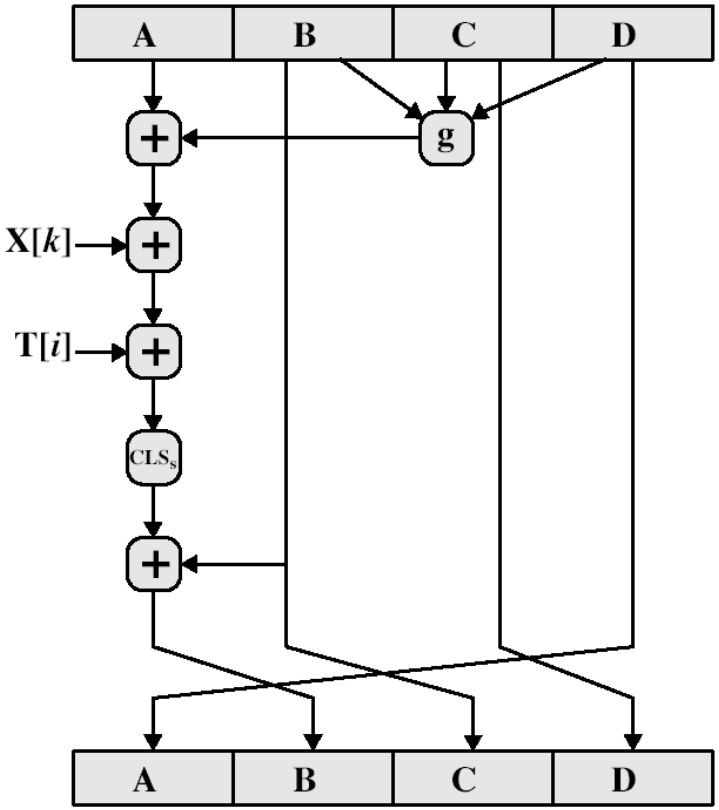
5.2 :Devide the current 512 bit block into 16 sub-blocks(size-32bits)

5.3: Now we have four rounds.Ineach round we process all the 16 sub-blocks belonging to a block

Conceptual process within a round



- In each round we have 16 input sub-blocks,named M[0],M[1],….,M[15].
- Also,t is an arrayof constants.It consists of 64 elements,with each element consisting of 32 bits.We denote the elemnts of this array t ast[0],t[1],……,t[63].
- **floor(2^{32} × abs (sin(i + 1))**


A
B
C
D
g
X[k]
T[i]
CLS_S
A
B
C
D

- Each round has 16 steps of the form:

```
a = b+((a+g(b,c,d)+X[k]+T[i])<<<s)
```

- g(d,b,c) = (b$\wedge$c)$\vee$(~b $\wedge$ d)
- g(d,b,c) = (b $\wedge$ d) $\vee$(c $\wedge$~ d)
- g(d,b,c) = b$\oplus$c$\oplus$ d
- g(d,b,c) = c$\oplus$(b $\wedge$ ~d)

How SHA-1 works?

- Step 1: Padding
- Step2 : Append length
- Step3 :Divide the input into 512-bit blocks
- Step4: Initialize chaining varaibles

A	Hex	01	25	45	67
B	Hex	89	AB	CD	EF
C	Hex	FE	DC	BA	98
D	Hex	76	54	32	10
E	Hex	C3	D2	E1	F0

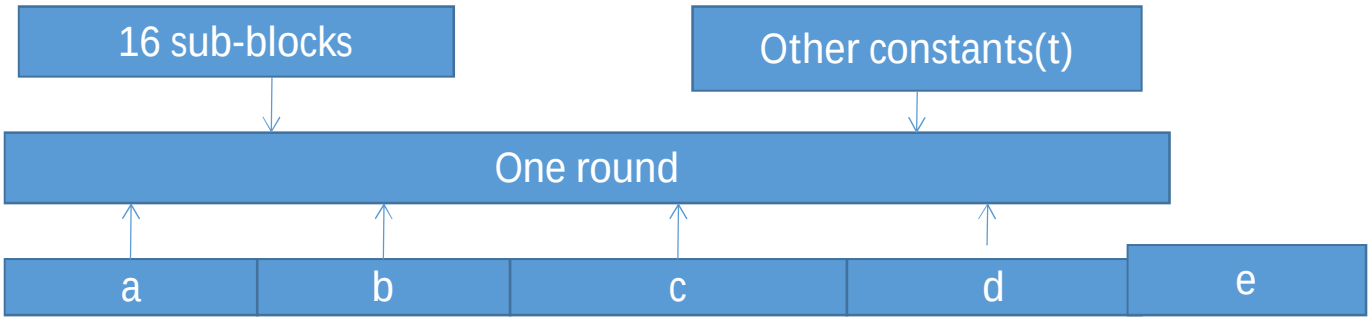
- Step 5: Process blocks

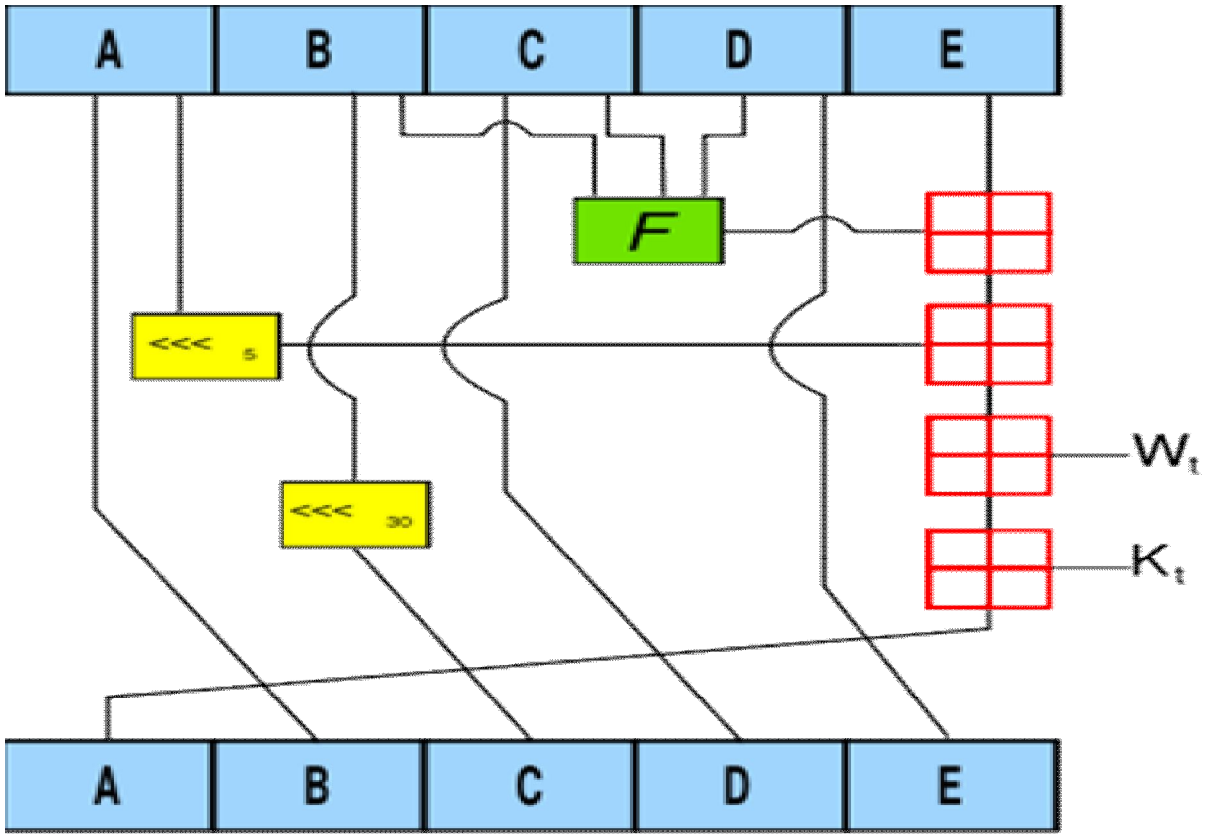
5.1: copy the chaining varaibles into five correspobding varaibles a,b,c , d and e

5.2 :Devide the current 512 bit block into 16 sub-blocks(size-32bits)

5.3: SHA-1has four rounds.each round consists of 20 steps

Conceptual process within a round




A
B
C
D
E
F
<<< 5
<<< 30
W_t
K_t
A
B
C
D
E

- Each round has 20 steps of the form:

abcde=(e + function F+s^5(a)+W[t]+K[t]),a, s^{30}(b),c,d

Round 1: (b AND c) OR ((NOT b) AND (d))
Round 2: B XOR c XOR d
Round 3: (b AND c) OR (b AND d) OR (c AND d)
Round 4: B XOR c XOR d

The remaining 64 values are defined using the equation
W[t]=s^1(W[t-16] XOR W[t-14] XOR W[t-8] XOR W[t-3])

Round	Value of t between	K[t] in Hex
1	1 and 19	5A 92 79 99
2	20 and 39	6E D9 EB A1
3	40 and 59	9F 1B BC DC
4	60 and 79	CA 62 C1 D6

How SHA-512 works?

- Step 1: Padding
- Step2 : Append length
- Step3 :Divide the input into 1024-bit blocks
- Step4: Initialize chaining variables

A 6A09E667F3BCC908

B BB67AE8584CAA73B

.....

H 5BE0CD19137E2179

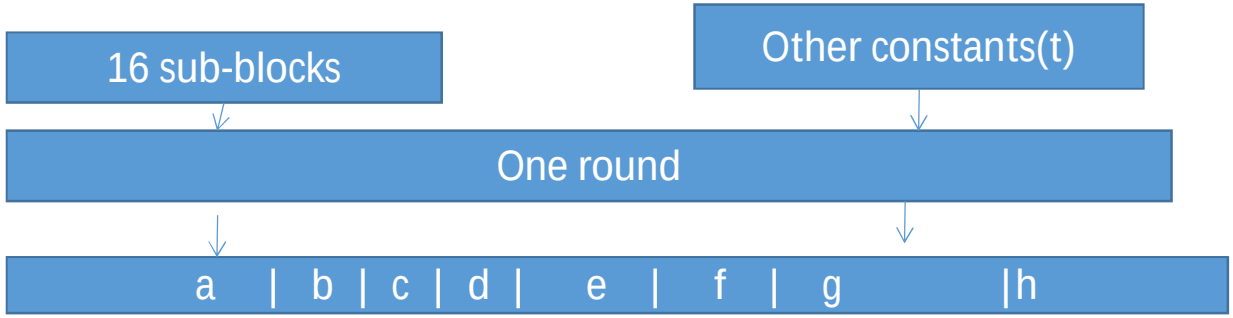
- Step 5: Process blocks

5.1: copy the chaining varaibles into four correspobding varaibles a,b,c and d

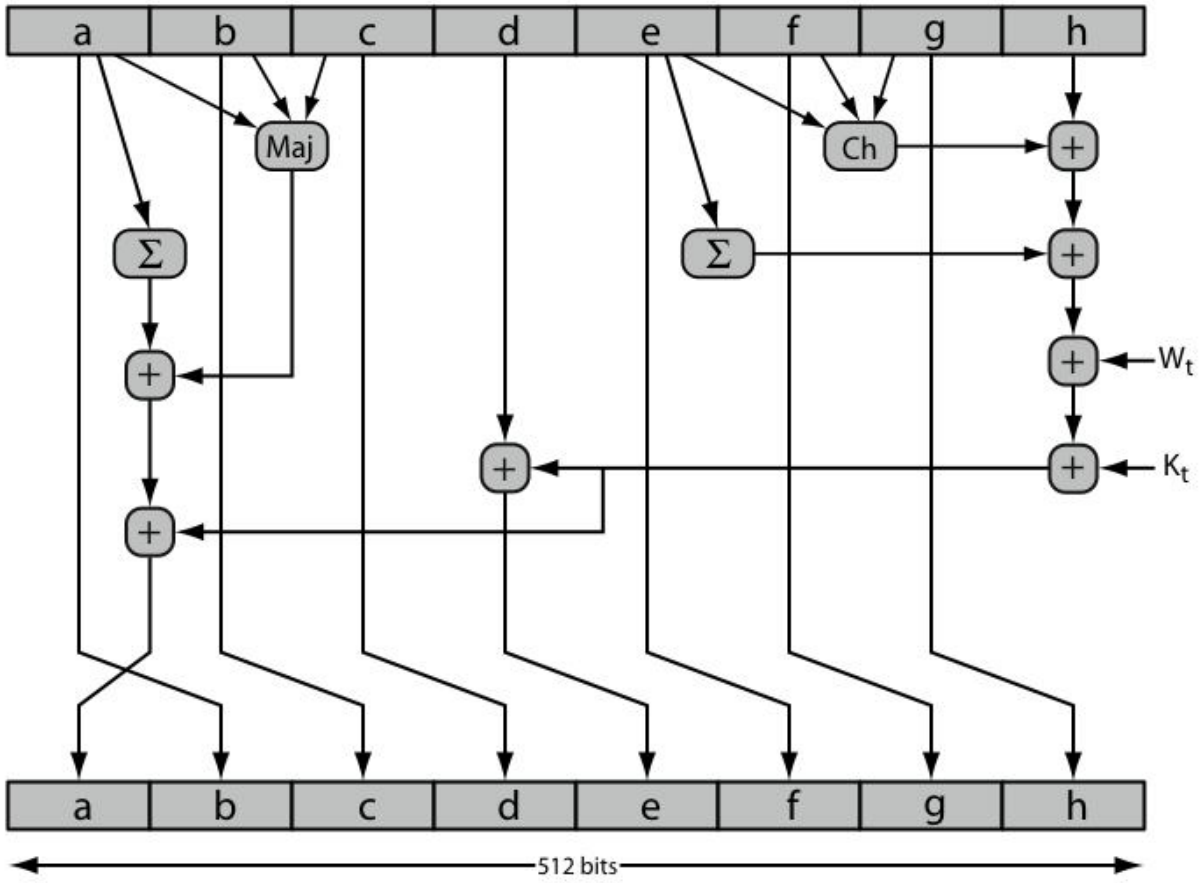
5.2 :Devide the current 1024 bit block into 16 sub-blocks(size-64 bits)

5.3: Now we have 80 rounds.In each round we process all the 16 sub-blocks belonging to a block

Conceptual process within a round



SHA-512 Round Function



Ch(e,f,g)=(e AND f) XOR (NOT e AND g)

Maj(a,b,c)= (a AND b) XOR (a AND c) XOR (b AND c)

Sum(a)= ROTR(a by 28 bits) XOR ROTR(a by 34 bits) XOR ROTR(a by 39 bits)

Sum(e)= ROTR(e by 14 bits) XOR ROTR(e by 18 bits) XOR ROTR(e by 41 bits)

W[t]=64-bit word derived from the current input block

K[t}=constants given in the book

Add= addition mod 2^{64}

1. **For the first 16 rounds (0 to 15),the value of W[t] is equal to the corresponding word in the message block.**
2. **For the remaining 64 steps,the value of W[t] is equal to the circular left shift by one bit of the XOR of the four preceding values of W[t] with two of them subjected to the circular left shift by 1 bit.**