SMVS
Page No.
Date : / /

Authentication Techniques

* Authentication Requirements:

This are the requirements of Authentication.

1 Release of message Contents: This process involves decrypting the ciphertext into its original form using the appropriate key and algorithm.

2 Traffic Analysis: Find the pattern of traffic between two system, the number and length of shared data between two parties.

3 Masquerade: Insertion of data into the network from a Fraudulent source.

4 Content Modification : Changes to the content of a data including modification of data.

5 Sequence Modification : Any modification to a sequence of message between two parties include modification of data.

6 Timing Modification : In this an entire session or individual messages in the sequence could be delayed or replayed.

7 Repudiation : Denial of transmission of message by source.

* Authentication Function :

There are Three Types of Three ways to produce Authentication Function.

It is also known as Types of

Functions to produce message Authentication.

This are methods to Produce Authentication.

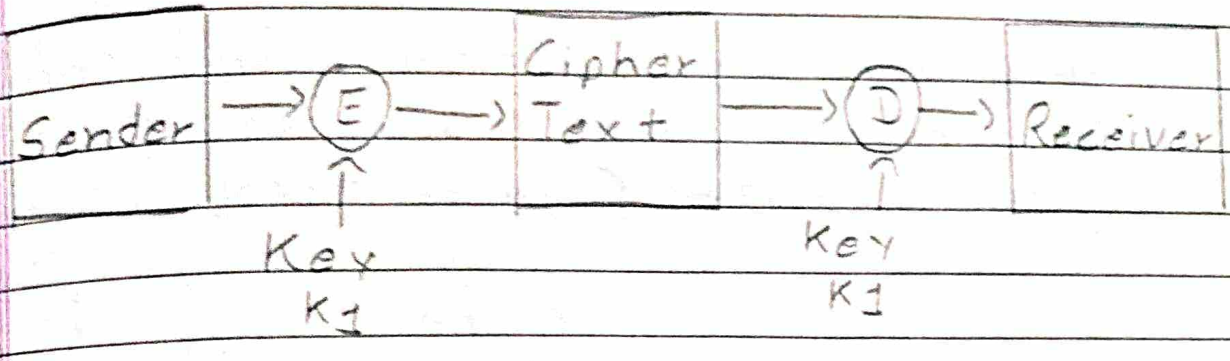
(a) Message Encryption
(b) Message Authentication
(c) Hash Functions.

(a) Message Encryption:

There are Four ways to provides Message Encryption

(1) Symmetric Encryption
(2) Asymmetric Encryption
(3) Public-key Encryption
(4) Public-key Encryption - confidentiality, Authentication and signature

(1) Symmetric Encryption: Confidentiality and Authentication



If Sender want to send the data to the Receiver than sender perform encryption using some algorithm and Key(K1) and create cipher text.

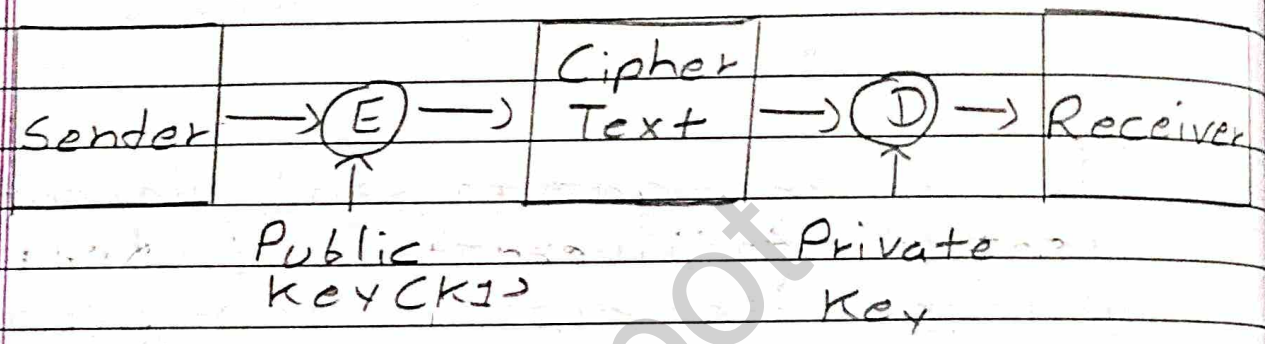
After that This Same Key(K1) is used for decryption for conformed that the Plain text is same as cipher text.

Using This way Confidentiality and Authentication is done.

(2) Asymmetric : Public Key Encryption: confidentiality.

If Sender want to send the data to the Receiver than sender

perform encryption using some algorithm and Public key (K1) and create Cipher text.



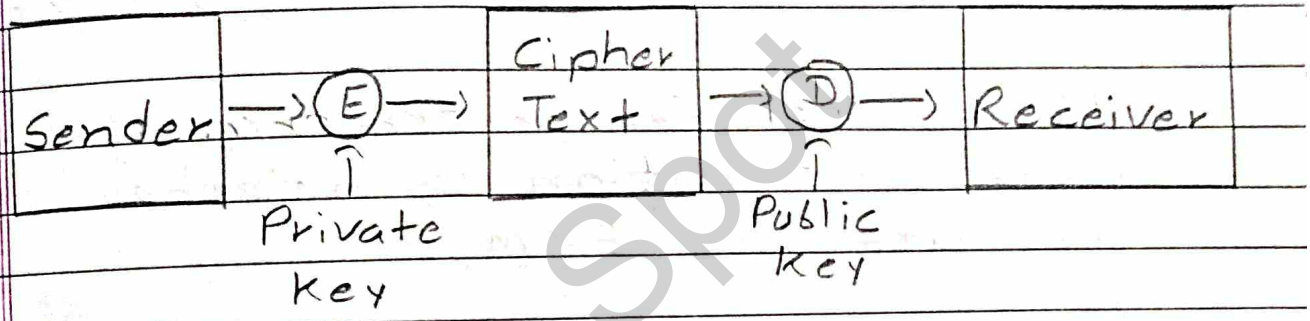
After that Receiver have to Perform Decryption for verify the Plain text using some algorithm and Private key and get plain text.

Here, We can done ~~Au~~ Confidentiality but No Authentication.

(3) Public-key Encryption : Signature and Authentication.

In this method, we will get the Authentication and not confidenticality.

If Sender want to send the data to the Receiver than sender perform encryption using some algorithm and Private Key and get cipher Key.



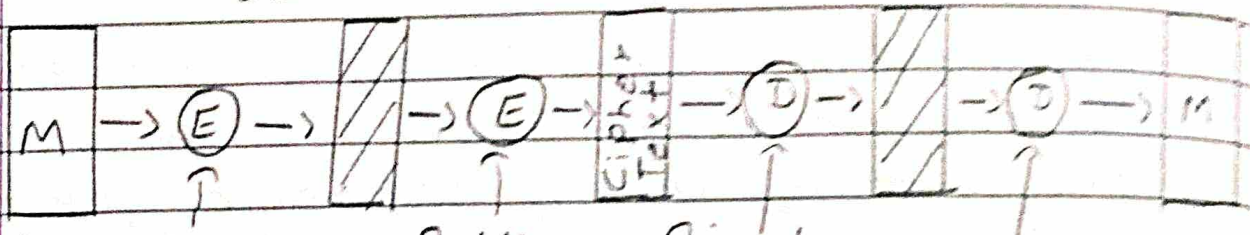
After that Receiver have to Perform Decryption for verify the Plain text using some Algorithm and Public Key and get plain text.

(4) Public-key Encryption : Confidentiality, Authentication and Signature.

To achieve both Authentication and Confidentiality, we have to perform Two times encryption and Decryption.

E(M, P_A)

D(M, P_B)



Sender Private Key P_A

Public Key

Private Key P_B

Public Key

Here, First Sender Perform the data encryption using the Private key E(M, P_A).

After that, Again we have to perform Encryption using the Public key and get Cipher Text.

After that, we have to perform Decryption using the private key D(M, P_B).

And Again, Perform Decryption, using the public key for get the plain text.

(6.2) Message Authentication:

In this, we have to use secret key to generate a small fixed size block of data called MAC or Cryptographic Checksum.

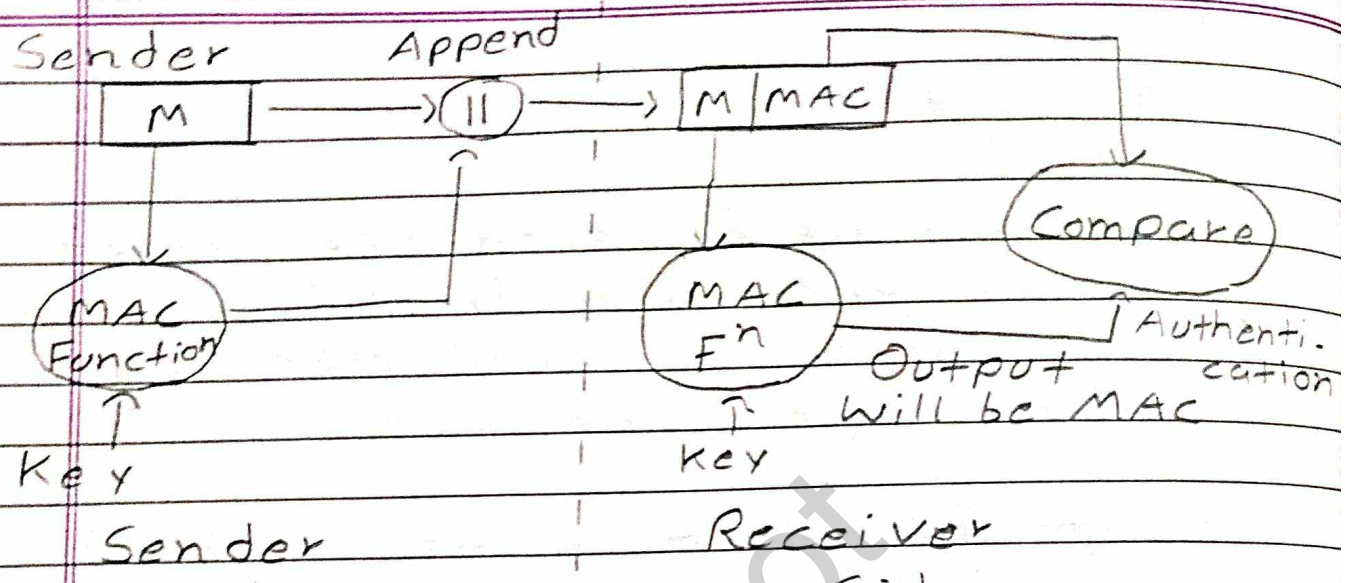
Let Sender A want to share the data to the Receiver B

Then, Calculation of MAC is

$MAC = C(K, M)$

Where, M = Input data which is shared by Sender.
C = MAC Function
K = Shared Secret Key.

An alternative to authentication we have to use secret key to generate a small Fixed block of data that appended to the data.



=> Sender Side:

Sender have to passed this Message into MAC Function.

Here, Shared Secret Key has by both sender and receiver

MAC Function is Generate output which is know as MAC.

After that MAC output and Original message is append(||).

=> Receiver Side:

At Receiver Side, we will separate the part of Message from (Message + MAC).

The message is passed in MAC Function which is create fixed size of block using the secret key.

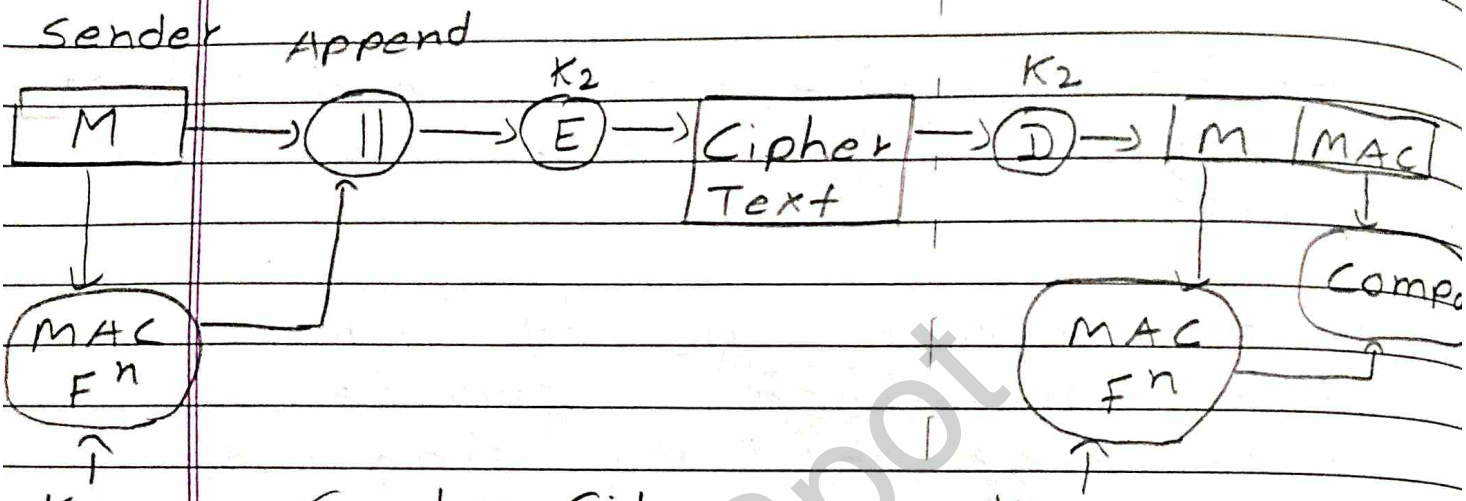
After the MAC output and MAC will be compare and so, it is known as authentication.

=> MAC For Authentication and Confidentiality:

There are Two ways to provides Authentication and Confidentiality

(i) Authentication tied to Plain text.
(ii) Authentication tied to Cipher text.

(i) Authentication tied to Plain text:



=> Sender Side:

Sender have to passed this Message into the MAC Function.

Here, Secret Key is shared between Sender and Receiver.

MAC Function is Generate output which is known as MAC.

After that MAC output and Original Message is append (||).

After append, Using Key, we have to perform encryption and get cipher text.

=> Receiver Side:

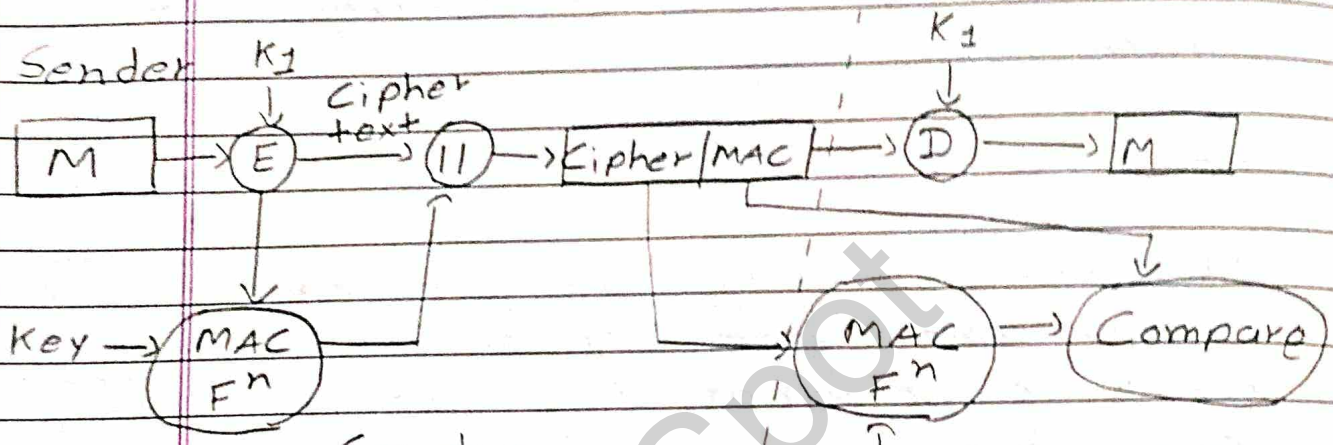
At Receiver Side, We get the Cipher text From Sender and we have to perform Decryption using Key which used in Encryption.

After Decryption, We have to separate the part of message From (Message + MAC).

Message will be passed into MAC Function and MAC will be passed into compare.

MAC Function Output will be compare to MAC.

(iii) Authenticate tied to Cipher text:



=> Sender Side:

Apply Message + Key in the encryption algorithm and user will get output.

This output + Key is given to MAC Function and in output MAC we will get.

=> Receiver Side:

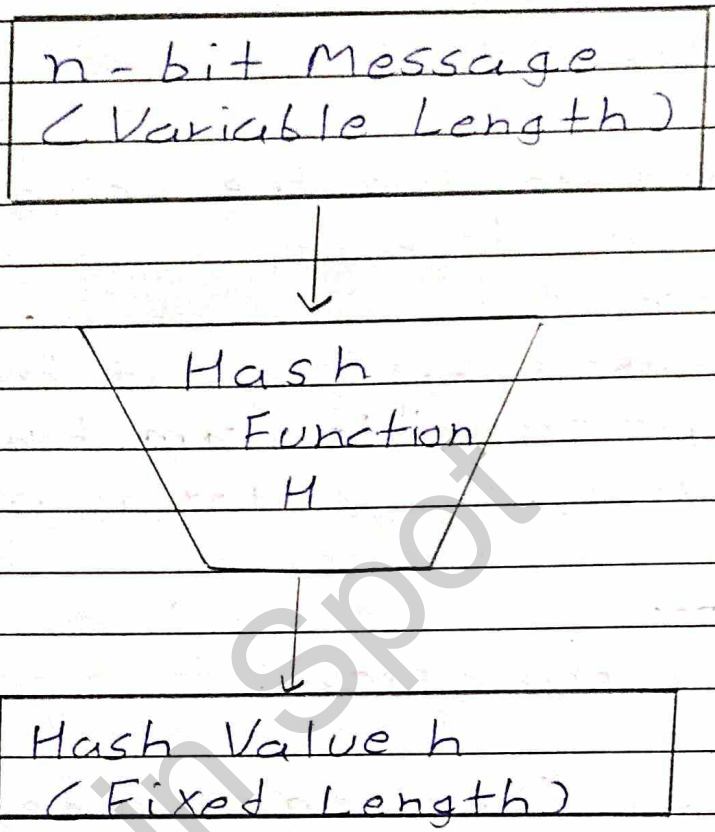
At Receiver side the value we need to decrypt.

So, for this we pass it from the decryption algorithm for this key used.

(c) Hash Function:

A Hash Function H accepts a variable-length block of data M as input and produces a fixed-size hase value h = H(M).

A Good hash Function has the property that the results of applying the function to a large set of inputs will produce output



There are Four Method, For use hash code for provide the message authentication.

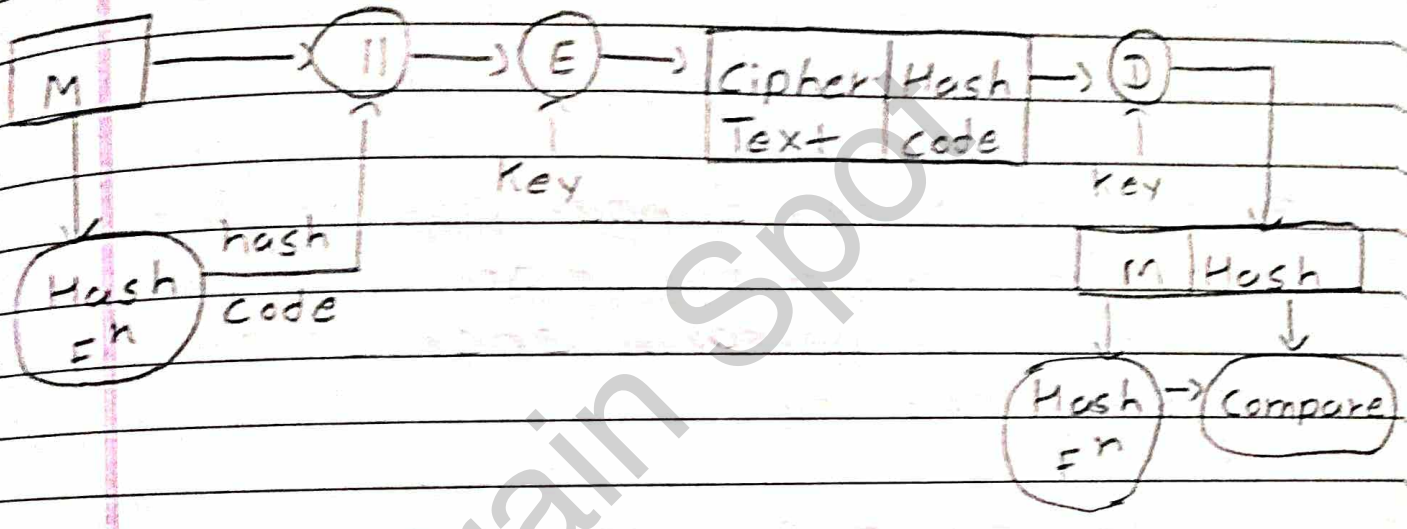
(i) Method 1 :

=) Server Side : Sender Original data is send to the Hash Function and create hash code.

After that hash code and Original data is appended

and using key we have to perform encryption and create Cipher text.

Sender Append



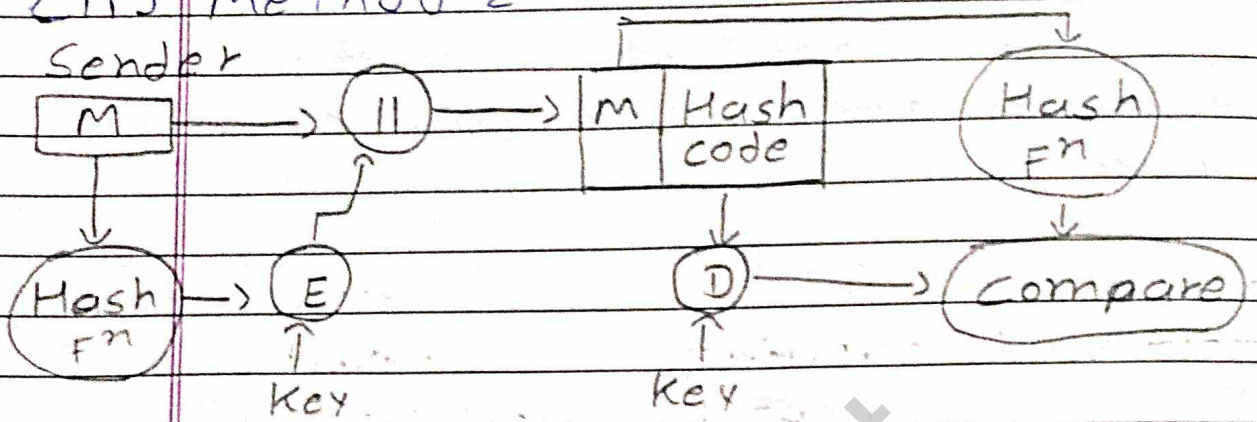
=> Receiver Side:

At Receiver Side, we have to perform Decryption using key and sepracte Message and Hash Code

After that Message is passed into Hash Function

and Hash Function O/P and Hash code will be compare.

(iii) Method 2:



=> Server Side: Sender passed data to the hash function and this hash code perform encryption using the key.

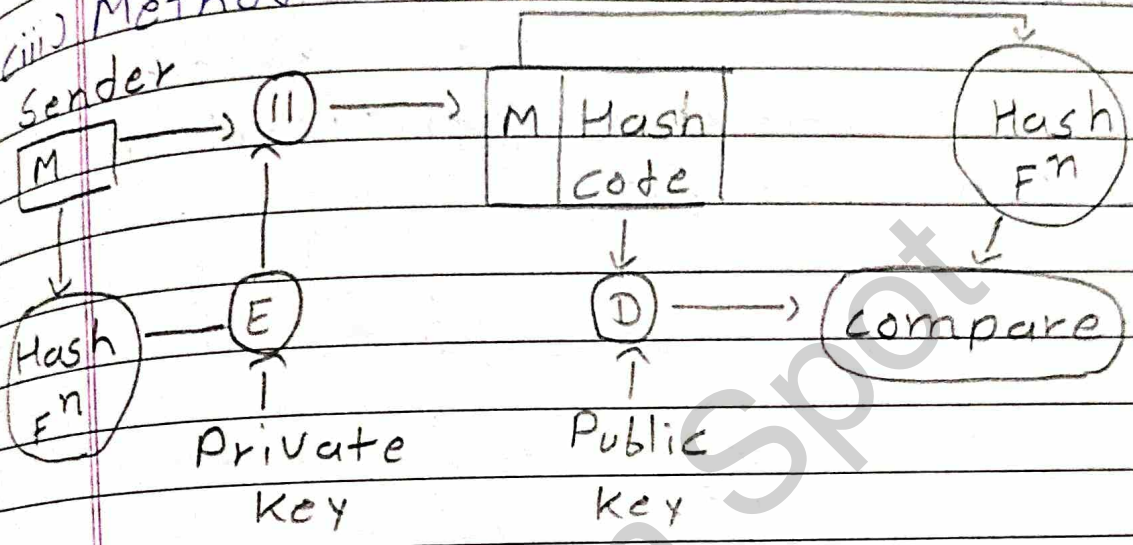
After that encrypt data and original data is append.

=> Receiver Side: Append data is divided into two part data and hash code.

Hash code perform decryption using key and goto the compare. and data is pass into Hash Function and goto the compare

Here, We achieve Authentication and No confidentiality.

(iii) Method 3:



=> Server Side: Sender passed data to the hash function and this hash code perform encryption using private key.

After that encrypt data and original data is append.

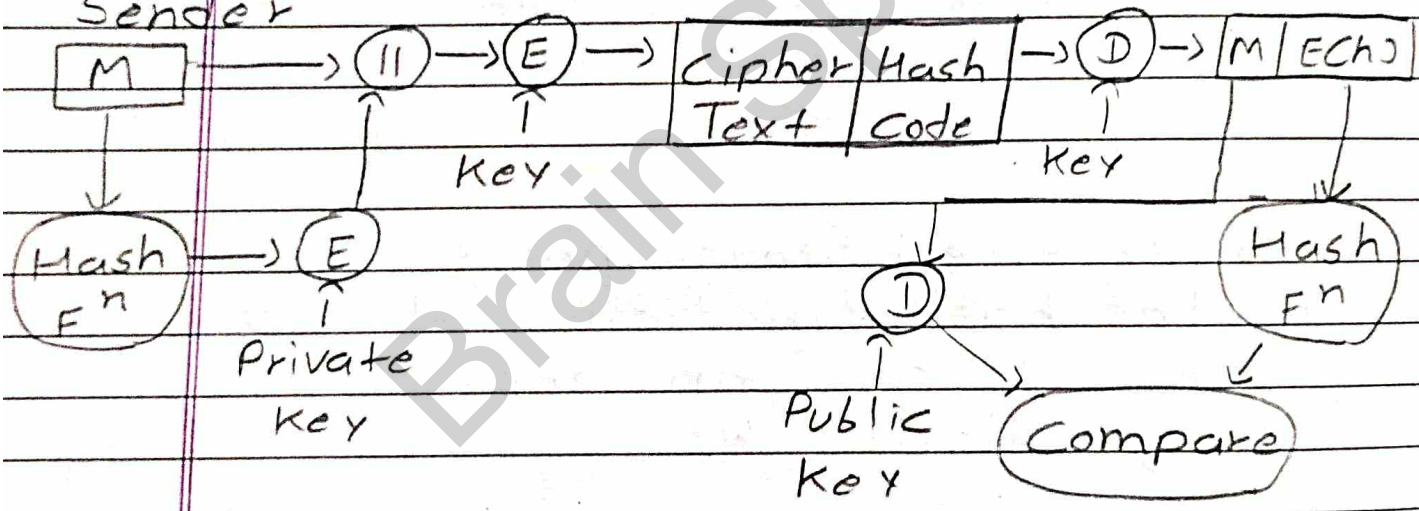
=> Receiver Side: Append data is divided into two part data and hash code.

Hash code perform decryption

using public key and goto the compare and data is pass into hash function and goto the compare.

Here, We achieve Authentication, No confidentiality.

(iv) Method 4:



=> Server Side: Sender passed data to the hash function and this hash code perform encryption using public key.

After that Original data and

Encrypt data is append. Append data is encrypt using key and divided into two part Cipher text and Hash code.

=> Receiver Side : Append Encrypt data is perform decryption using key and divided into two part original data and Hash code.

Hash code is passed into Hash function and goto the compare, and original data is perform the decryption using public key and goto the compare.

* Security of Hash Function and MACs.

There are Two Type of Attack is done on MACs.

(i) Brute - Force Attack : Strong collision resistance hash have cost $2^{m/2}$

Attacker can either attack keyspace.

(ii) Cryptanalytic Attacks: Like Block ciphers want brute-force attack to be the best alternative.

Have a number of analytic attacks on iterated hash function.

* Message Digest Algorithm:

It is a cryptographic hash Function Algorithm.

It will takes Any size of input message and will give fixed size of message digest.

In I/P you can take any length of code but in output you will get 128-bit size digest

The Purpose of this algorithm is to check the Integrity means our message is same or not when we transfer data from one computer to another computer.

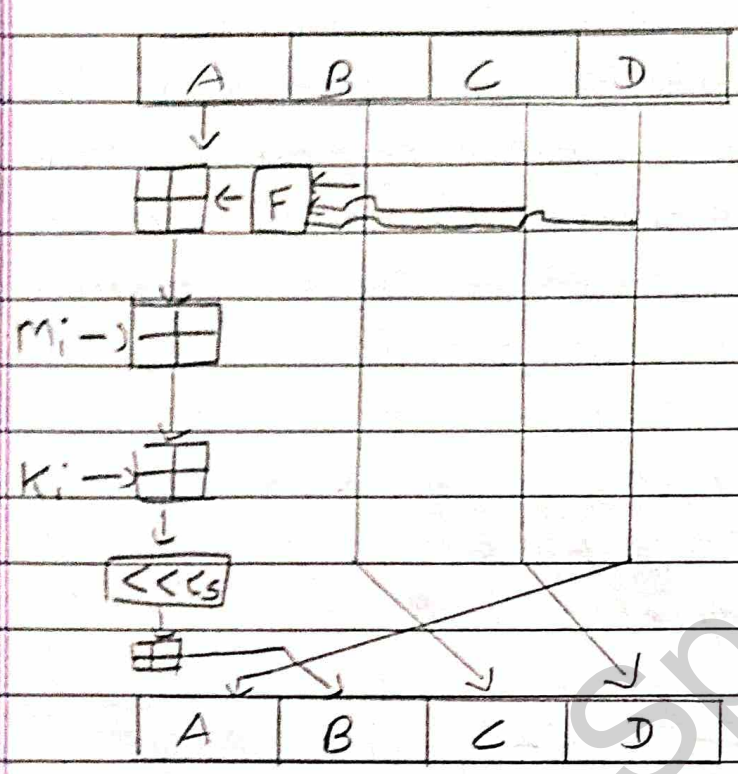
The Size of message block
size = 512 bits
Number of Rounds = 4

=> There are Four Step in MD5:

(i) Append Padding Bits
(ii) Append 64-bit Representation
(iii) Initialize MD Buffers
(iv) Process Each Block.

(i) Append Padding Bits: In this Algorithm I/P block size is not Fixed but O/P block size is Fixed.

Suppose, Original message have 1000 bits than we need to add extra bits.



We have to add extra bit such way that can gives multiple of - 64 bits.

512

So, For 1000 bits,

1536 - 64

512 X 3 - 64 = 1472

So, We have to add 1472 - 1000 = 472 extra bits.

(iii) Append 64-bits Representation:

In this step, we have to divided the Bits into 512 bits block.

For 1000, We get 1536 bits,
So, We can divided in 3 blocks
512 512 512

(iii) Initialize MD Buffers: In MD5 the buffers as A, B, C and D are 32 bits size block.

So, Total O/P = 32 * 4 = 128-bits.

(iv) Process Each Block: In MD5, there are 4 Rounds and For every rounds we have to perform 16 operation

For 4-Blocks = 16 x 4 = 64 Rounds.

In Operation, we have to use One Non-Linear Function, For 4 Round we have to use 4 different Functions.

SMVS
Page No.
Date : / /

$\boxplus$ -> Addition Modulo, it will use 2^{32}

M_i -> Denotes 32-bit block of I/P

K_i -> It Represent 32-bit constant this represents LCS.

=> Working of MD5 :

For every Round, we have to perform 16 operation.

For Round,
1st Round Function = F
2nd Round Function = G
3rd Round Function = H
4th Round Function = I

We have to perform OR, AND, XOR and NOT for calculating Function.

$F(B, C, D) = (B \wedge C) \vee (\neg B \wedge D)$
$G(B, C, D) = (B \wedge D) \vee (C \wedge \neg D)$
$H(B, C, D) = B \oplus C \oplus D$

SMVS
Page No.
Date: / /

I(B, C, D) = C ⊕ (B ∨ ¬D)

After calculate Function, We perform an operation on each block.

- add Modulo 2^{32}
- M[i] -> 32 bit data
- K[i] -> 32 bit constant
- <<< n -> Left shift by n bits.

In the First Step, Output of B, C and D are taken and then the Function F is applied to them.

We will add modulo 2^{32} bits for the output of this with J.

We add the M[i] bit data with the output of the First step.

Then add 32-bits constant with the output of the second step.

At Last, we do Left shift operation by n.

After all steps, The Result of B will be Fed into C, Now same steps will be used For all Function.

=> Disadvantages:

1 MD5 Generates the same Hash Function For different inputs.

2 Poor Security

3 Insecure Algorithm.

=> Application:

1 Provides Secure Password For Users.

2 It is used For File Authentication.

3 Used to verify the data integrity.

SMVS
Page No.
Date : / /

* Secure Hash Algorithm:

Secure Hash Algorithm is a modified version of MD5 Algorithm.

In this Algorithm, Input Block size is variable and size of output block is 160 bits.

Secure Hash Algorithm 1 is a hash function which takes an input and produces 160-bits hash value.

There are Five steps in SHA:

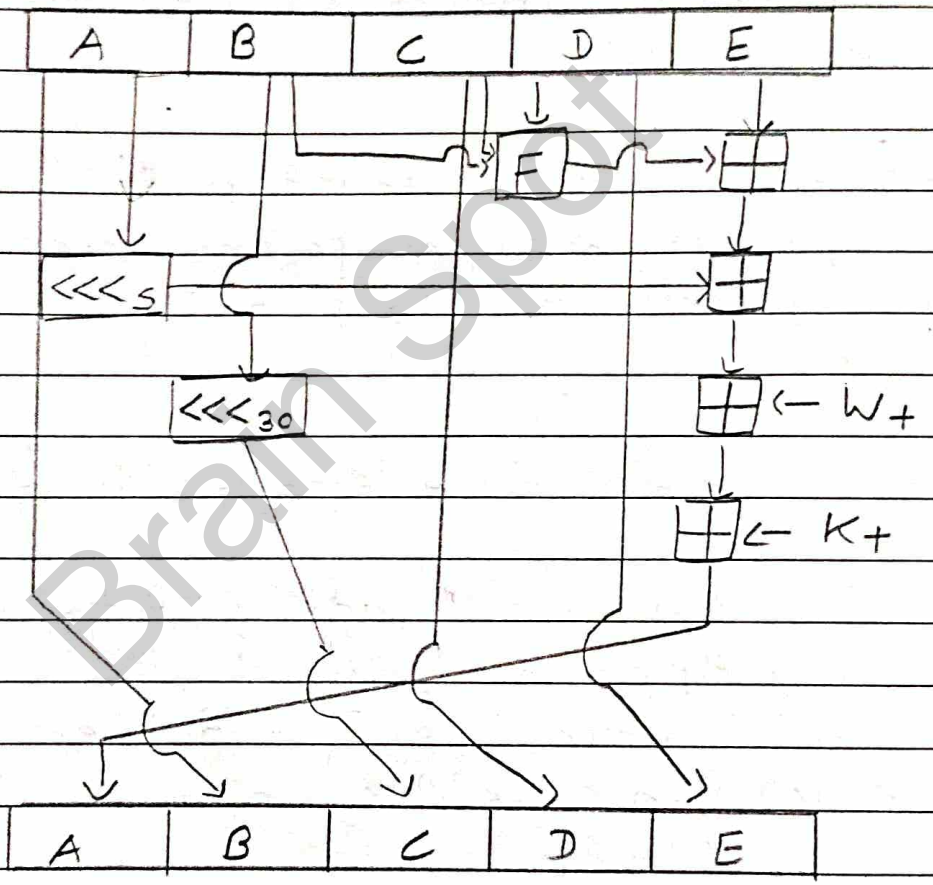
(i) Append Padding Bits
(ii) Append Length
(iii) Initialize The Buffer
(iv) Process Message in 512-bits blocks
(v) Output.

(i) Append Padding Bits: In SHA, According to MD5 Algorithm we have to add bits.

We have to ~~bit~~ add bit in such a way that can gives,

Multiple of - 64 bits.

512



(iii) Append Length: A 64-bits block considered as an unsigned 64-bit Integer and defining the length of the original message is added to the data.

(iii) Initialize The Buffer: In SHA, we have 5 buffer as A, B, C, D and E

The Buffer includes 5 registers of 32-bits which gives 160-bits of output

These Five Register are initialized to the following 32-bits Integers.

A = 67 45 23 01
B = ef cd ab 89
C = 98 ba dc fe
D = 10 32 54 76
E = c3 d2 e1 f0

(iv) Process Message in 512-bits Blocks:

The compression function is divided into 20 sequential steps, For each round is made up of 20 steps.

For Every Round we have to use different Boolean function which

define as F1, F2, F3 and F4.

(v) Output : After processing the final 512-bit message block, it can obtain a 160-bit message digest.

* Digital Signature :

Digital Signature is a mathematical techniques that is used to ensure the authenticity and Integrity of a message or data.

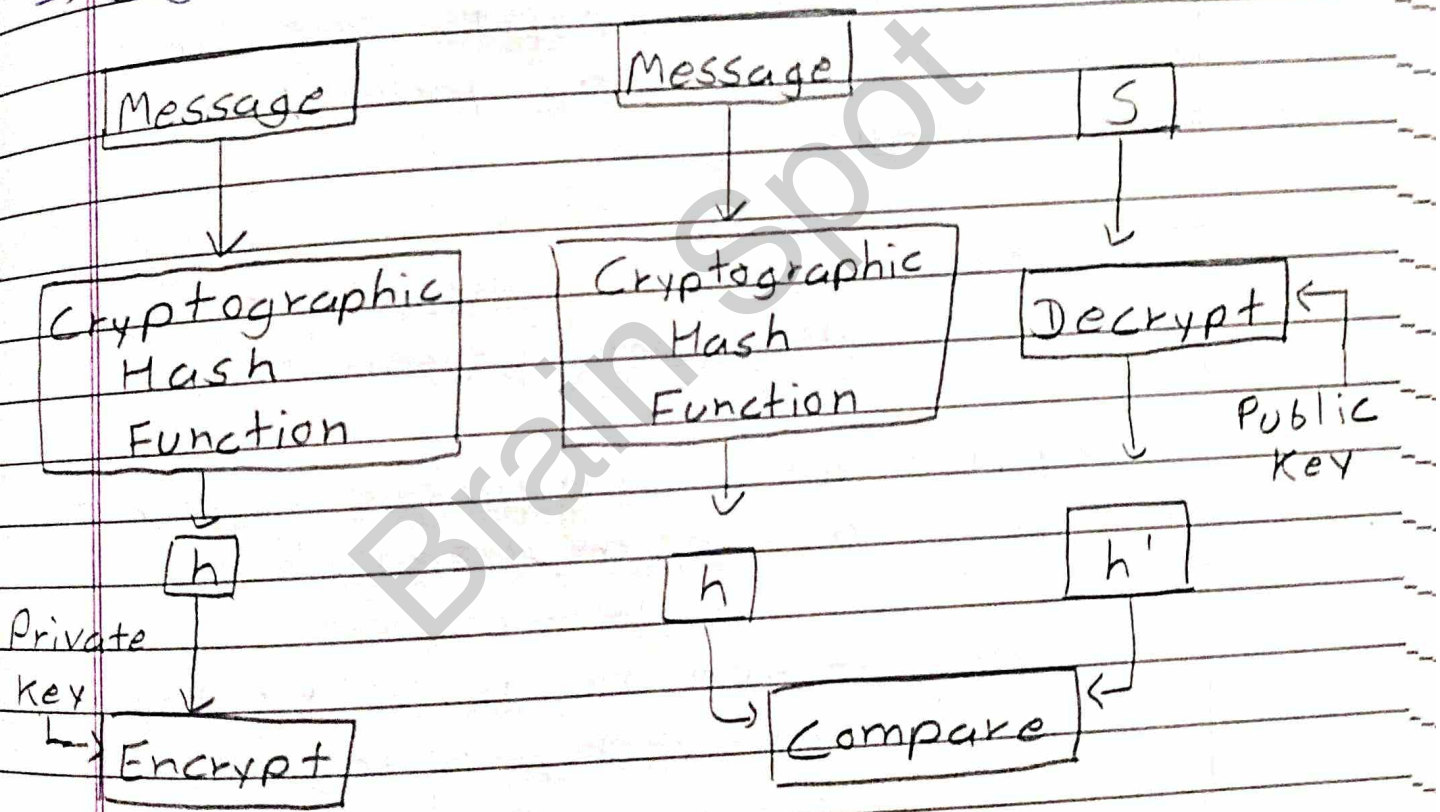
It is similar to the signature which is made by hand.

It ensures that the data or any other electronic document is original and Document is not altered.

It is based on the method of Public Key Asymmetric cryptography

Two keys are used in this. The Public key is used to encrypt the data and Private key is used for decryption.

=> Digital Signature Mechanism:



Steps:

1 Select a File to be digitally signed.

2 Using Hash Function, calculate hash value. This File is encrypted

By using Private Key of Sender.

3 The original file content along with the digital signature is transmitted.

4 The Receiver decrypts the digital signature by using public key of a sender.

5 The Receiver now has the file content and can compute it.

6 Compare this computed file with the original computed file.

The comparison needs to be the same for ensuring integrity.

→ Application of Digital Signature:

(a) Authentication: In the Digital Signature, Authentication helps to authenticate the sources of data.

(b) Non-Repudiation : It ensures that their signature on a document or in a file that cannot be denied.

(c) Integrity : It ensures that the data is read, accurate and not modify by unauthorized user.

-> Benefits of Digital Signature :

(a) Legal Document and contracts : Digital Signature makes them ideal for any legal document that need authentication.

(b) Sales Contracts : Digital signing for contracts and sales contracts authenticates the identity of the seller and the Buyer.

(c) Health Data : Digital Signature ensure that the Health document is confidential.

-> Disadvantages:

a) Dependency on Technology:

b) Complexity:

c) Limited Acceptance:

* Digital Signature standard:

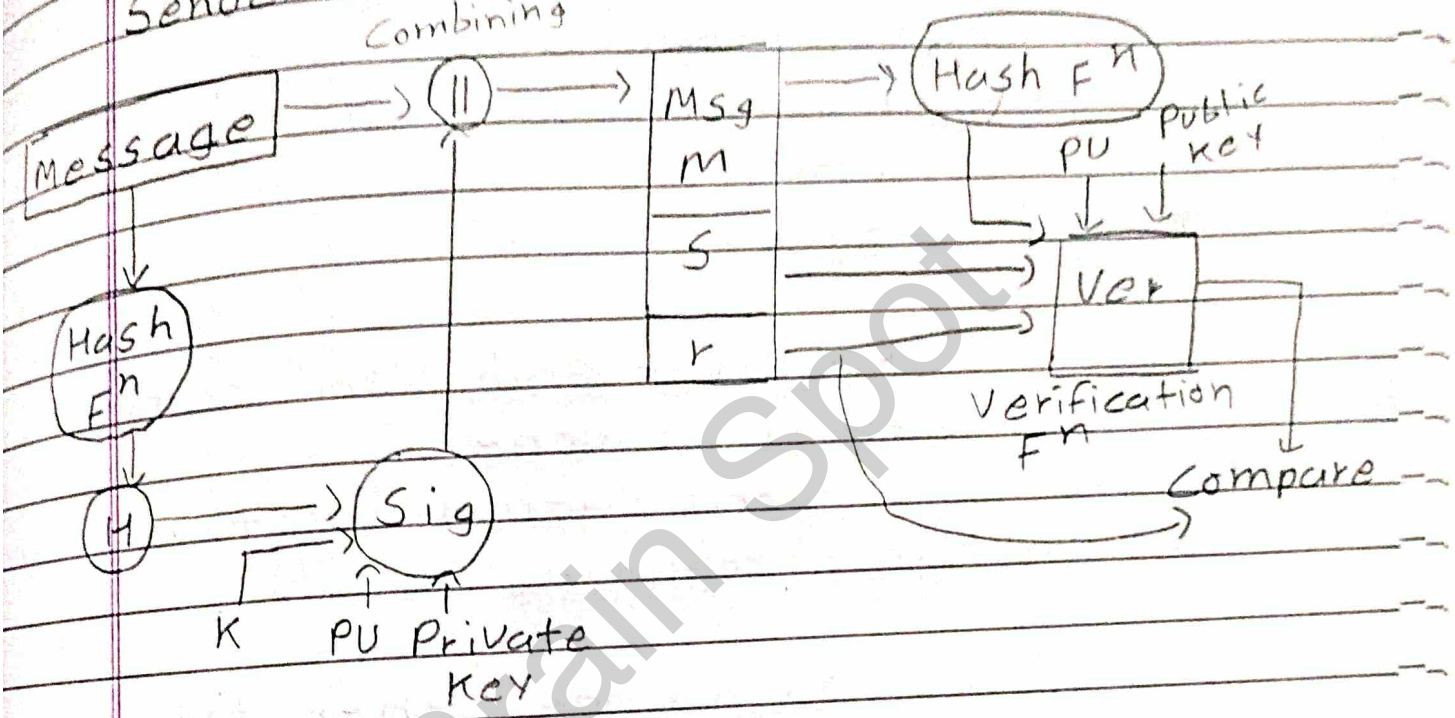
The Digital Signature Standard makes use of the SHA.

The DSS uses an algorithm that is designed to provide only the Digital Signature Function.

DSS is a Federal Information Processing Standard which defines algorithms that are used to generate digital signature.

DSS only provides us with the Digital Signature Function not

with any encryption.



⇒ Sender Side :

In DSS Approach, A hash code is generated out of the message. Following inputs are given to the Signature Function :

(1) Hash Code
(2) The Random Number 'K' generated For Signature.

(3) The Private Key
(4) A Global Public Key - This input function is provides output signature two components 's' and 'r'.

=> Receiver Side:

At the Receiver end, verification of the sender is done.
There is a Verification Function which takes following inputs:

(1) The Hash Code generated by the Receiver
(2) Signature components 's' and 'r'
(3) Public Key of the Sender
(4) Global Public Key:

The output of the verification function is compared with 'r'.

Both the values will match if the

sent is signature is valid.

* Difference between MD5 and SHA:

		MD5	SHA
1	Length in bits	128	160
2	Attack to Find Original Msg	2^{128} Operation	2^{160} Operation
3	Two Msg. with same MD	2^{64} Operation	2^{80} Operation
4	Speed	Faster	Slower
5	Successfull Attacks	Break MD5	No Such claim

SMVS
Page No.
Date : / /

* Difference between RSA and DSS

	RSA	DSS
1	Cryptosystem Algorithm	Digital Signature Algorithm.
2	Used For Secure Data transmission.	Used For Verify data.
3	Developed in 1977	Developed in 1991
4	Uses Mathematical concepts.	Uses Discrete Algorithm
5	Key Generation is slow.	Key Generation is Faster.
6	Faster than DSA	Slower
7	Slower in decryption	Faster in encryption.