Jaipur Engineering College & Research Centre, Jaipur
Department of Computer Science & Engineering

Information Security System
[6CS4-03]
Notes

Prepared By:

Kanishk Jain
Ashish Ameria

Assistant Prof., CSE

VISION AND MISSION OF INSTITUTE

VISION

To become renowned centre of outcome based learning and work towards academic, professional, cultural and social enrichments of the lives of individual and communities"

MISSION

M1. Focus on evaluation of learning outcomes and motivate students to inculcate research aptitude by project based learning.

M2. Identify areas of focus and provide platform to gain knowledge and solutions based on informed perception of Indian, regional and global needs.

M3. Offer opportunities for interaction between academia and industry.

M4. Develop human potential to its fullest extent so that intellectually capable and imaginatively gifted leaders can emerge in a range of professions.

VISION AND MISSION OF DEPARTMENT

VISION

To become renowned Centre of excellence in computer science and engineering and make competent engineers & professionals with high ethical values prepared for lifelong learning.

MISSION

M1: To impart outcome based education for emerging technologies in the field of computer science and engineering.

M2: To provide opportunities for interaction between academia and industry.

M3: To provide platform for lifelong learning by accepting the change in technologies

M4: To develop aptitude of fulfilling social responsibilities.

COURSE OUTCOMES

On completion of the course, students will be able to:

CO1: Identify different security attacks, Mechanism, classical and modern encryption techniques.

CO2: Apply random number generation, AES and S-box theory and Implement public key cryptosystem.

CO3: Evaluate message authentication and digital signatures using hash function and IP security.

CO4: Analyze & Implement Water marking technique and strong password protocol in Information Security System.

PROGRAM OUTCOMES (PO)

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Educational Objectives (PEO)

1. To provide students with the fundamentals of Engineering Sciences with more emphasis in Computer Science & Engineering by way of analyzing and exploiting Engineering challenge

2. To train students with good scientific and engineering knowledge so as to comprehend, analyze, design, and create novel products and solutions for the real life problems.

3. To inculcate professional and ethical attitude, effective communication skills, teamwork skills, multidisciplinary approach, entrepreneurial thinking and an ability to relate engineering issues with social issues.

4. To provide students with an academic environment aware of excellence, leadership, written ethical codes and guidelines, and the self-motivated life-long learning needed for a successful professional career.

5. To prepare students to excel in Industry and Higher education by Educating Students along with High moral values and Knowledge.

MAPPING CO-PO

Cos/POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	1	1	1	1	1	1	3
CO2	3	3	3	3	2	1	1	1	1	2	1	3
CO3	3	3	3	3	2	1	1	2	1	2	1	3
CO4	3	3	3	3	2	2	2	2	1	2	1	3

Program Specific Outcome's (PSO)

PSO1: Ability to interpret and analyze network specific and cyber security issues, automation in real word environment.

PSO2: Ability to Design and Develop Mobile and Web-based applications under realistic constraints.

Syllabus

SN	Contents	Hours
1	**Introduction:** Objective, scope and outcome of the course.	01
2	**Introduction to security attacks:** services and mechanism, classical encryption techniques- substitution ciphers and transposition ciphers, cryptanalysis, stream and block ciphers.	06
3	**Modern block ciphers**: Block Cipher structure, Data Encryption standard (DES) with example, strength of DES, Design principles of block cipher, AES with structure, its transformation functions, key expansion, example and implementation. Multiple encryption and triple DES, Electronic Code Book, Cipher Block Chaining Mode, Cipher Feedback mode, Output Feedback mode, Counter mode.	06
4	**Public Key Cryptosystems with Applications:** Requirements and Cryptanalysis, RSA cryptosystem, Rabin cryptosystem, Elgamal cryptosystem, Elliptic curve cryptosystem.	06
5	**Cryptographic Hash Functions, their applications:** Simple hash functions, its requirements and security, Hash functions based on Cipher Block Chaining, Secure Hash Algorithm (SHA). Message Authentication Codes, its requirements and security, MACs based on Hash Functions, Macs based on Block Ciphers. Digital Signature, its properties, requirements and security, various digital signature schemes (Elgamal and Schnorr), NIST digital Signature algorithm.	05
6	**Key management and distribution:** symmetric key distribution using symmetric and asymmetric encryptions, distribution of public keys, X.509 certificates, Public key infrastructure. Remote user authentication with symmetric and asymmetric encryption, Kerberos Web Security threats and approaches, SSL architecture and protocol, Transport layer security, HTTPS and SSH.	04
	Total	28

LECTURE PLAN

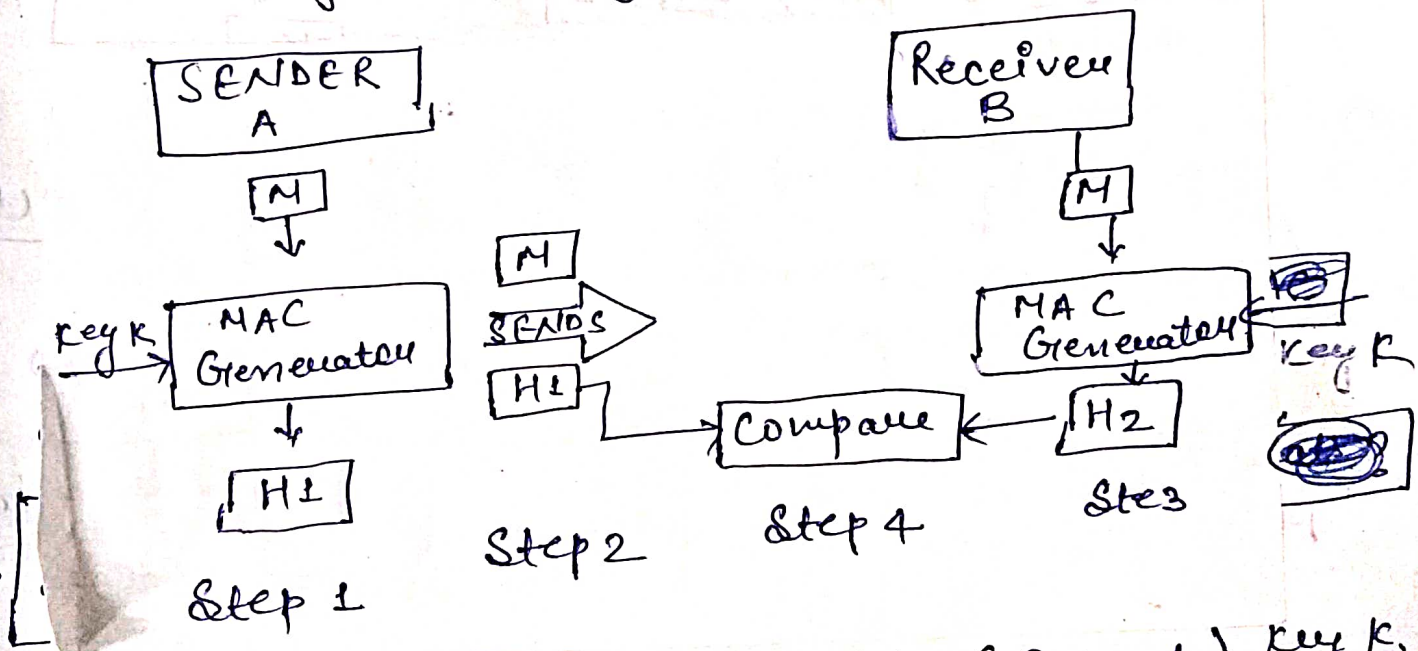
JAIPUR ENGINEERING COLLEGE AND RESEARCH CENTRE			
DEPARTMENT OF COMPUTER SCIENCE ENGINEERING			
LECTURE PLAN			
Subject: Information Security System (6CS4-03)		**Year/Sem: III/ VI**	
No. of Lecture Reqd./(Avl.) : 30 / 30			
Semester Starting:	**Semester Ending:**		
Unit No./ Total Lecture Reqd.	**Topics to be Delivered**	**Lect. Reqd.**	**Lect. No.**
Unit-1 (1)	Objective, Scope , Outcome of the course.	1	1
Unit-2 (6)	Introduction to security attacks	1	2
	services and mechanisms	1	3
	Classical encryption techniques	1	4
	substitution ciphers and transposition ciphers,	1	5
	crypt analysis	1	6
	Stream and block ciphers	1	7
Unit 3- (6)	Modern Block Ciphers: Block ciphers structure	1	8
	Data Encryption Standard(DES), Strength of DES	1	9
	Design principle of block cipher	1	10
	AES with Structure, Key Expansion	1	11
	Multiple Encryption and triple DES	1	12
	Cipher Block Chaining Mode, Cipher feedback mode, Counter mode	1	13
BC-1	**IDEA 64 Bit Encryption & MD5 Message Digest Algorithm**	1	14
Unit 4- (6)	Public Key Cryptosystems: Requirements	1	15
	Public Key Cryptosystems: Analysis	1	16
	RSA Cryptosystem	1	17
	Rabin Cryptosystem	1	18
	Elgamal Cryptosystem	1	19
	Elliptic Curve Cryptosystem	1	20
Unit 5- (5)	Cryptographic Hash Functions, Hash Function based on Cipher Block Chaining	1	21
	Secure Hash Algorithm	1	22
	Message Authentication Code	1	23
	MAC based on Hash Function & Block Cipher	1	24
	Digital Signature, Various Digital Signature Schemes, NIST Digital Signature	1	25
BC-2	**IP Security with Strong Password Protocols**	1	26
Unit 6- (4)	Key Management & Distribution, X.509 Certificates	1	27
	Remote User Authentication	1	28
	Web Security Threats, SSL Architecture	1	29
	Transport Layer Security, HTTPs & SSH	1	30
References:			
1) Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education			
2) Trappe & Washington, Introduction to Cryptography, 2nd Ed. Pearson.			
3) Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson			

Unit - IV

Message Authentication :- Msg authentication can be provided using the cryptographic techniques that use secret keys as done in case of encryption.

MAC algorithm is a symmetric key cryptographic technique to provide msg authentication. For establishing MAC process, The sender & receiver share a symmetric key K.

Message authentication is a service used to verify the integrity of message.

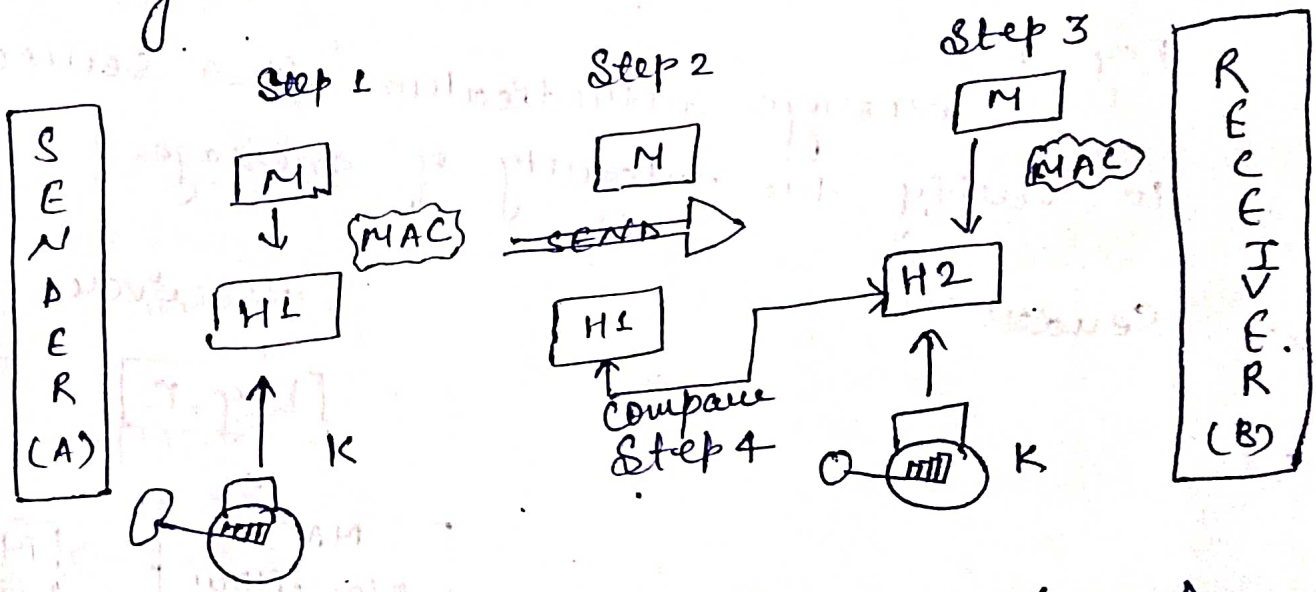


1. A & B share a symmetric (secret) key K, which is not known to anyone else. A calculate the MAC by applying key K to the message M.

2. A then sends the original message M and MAC H1 to B

3. When B receives the msg, B also uses K to calculate its own MAC H2 over M.

4. B now compares H1 with H2. If the 2 match, B concludes that the message M has not been changed during transit. However, if H1 ≠ H2, B rejects the msg, realizing that the message was changed during transit.

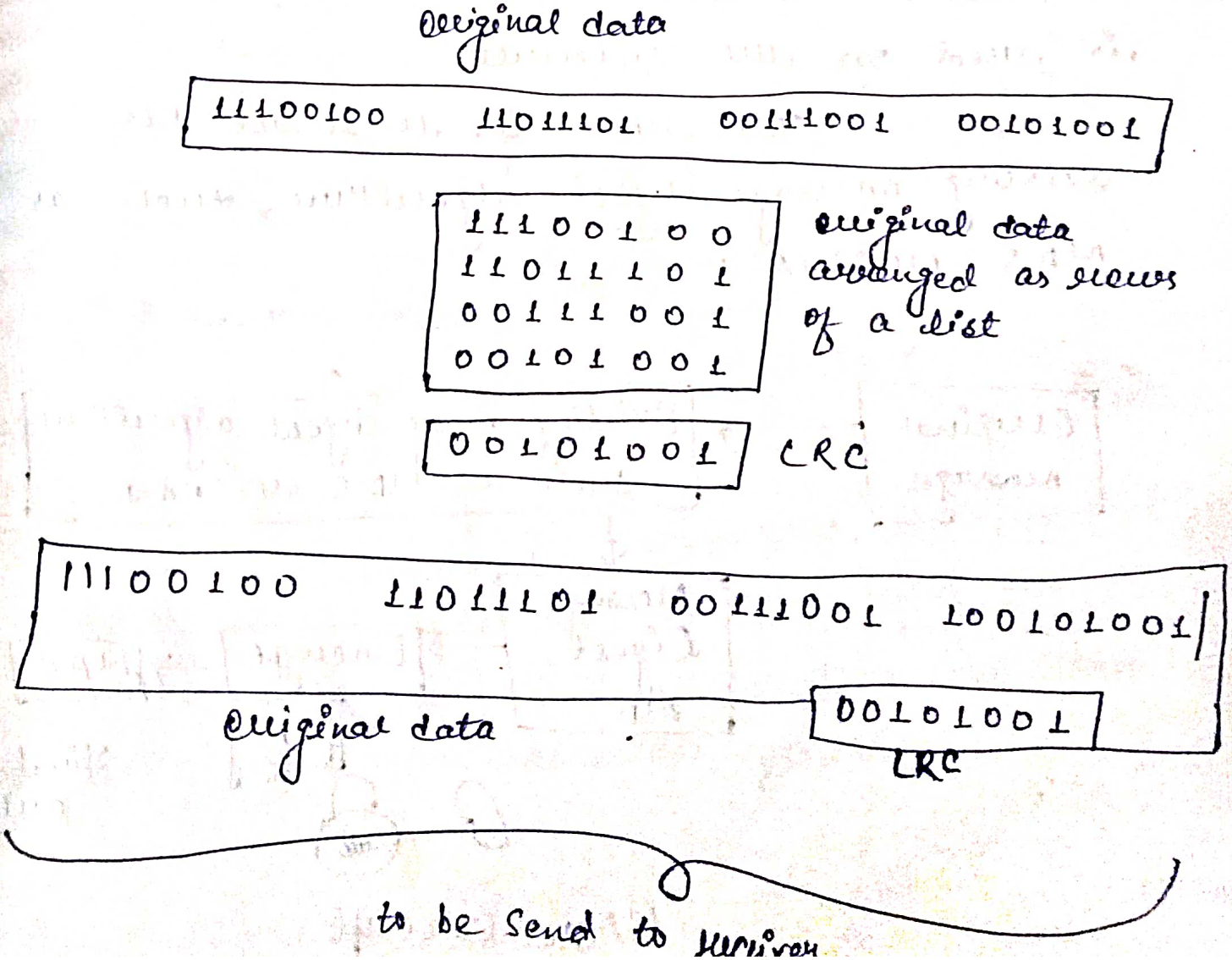


Message Authentication code (MAC)

Hash Functions :- Hash functions are the techniques that is used to generates the fingerprint or summary of a message that is known as message digest or hash code of a msg.

It is similar to the concepts like generating the CRC (cyclic Redundancy check) of message that is used to verify the integrity of data i.e. to ensure that a message has not been tampered after it leaves the sender but before it reaches the receiver.

Hash function is the variation of MAC. A hash function accept a variable size message M as input and produces a fixed size output known as hash code.

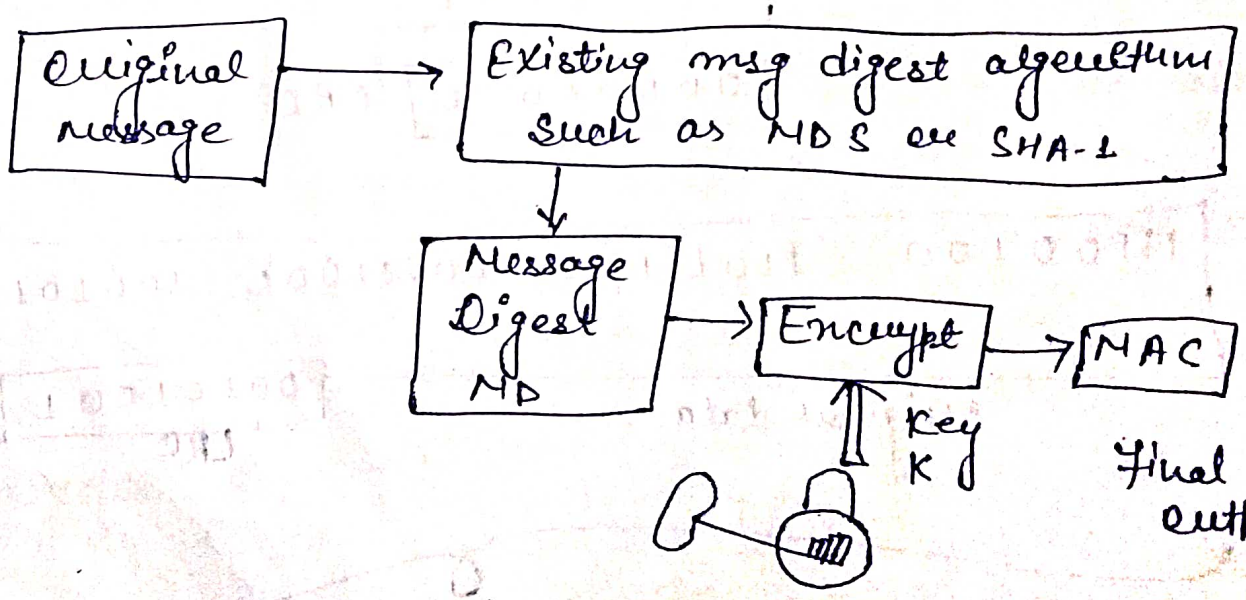


if the no. of 1's in the column is odd, then we say that the column has odd parity and it is indicated by 1, otherwise if the no. of 1's in the column is even it is indicated by 0 in the LRC calculation process.

HMAC → Hash based Message Authentication code :-

HMAC has been choosen as a mandatory security implementation for the Internet Protocol (IP) security and is also used in Secure Socket layer (SSL) Protocol, is used on the Internet.

This HMAC is to reuse the existing message digest algorithm, such as MD5 or SHA-1.

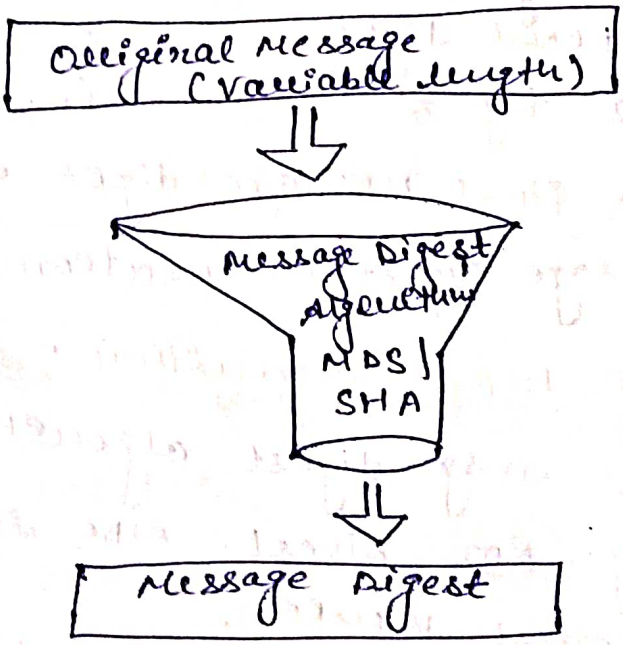


HMAC Concept

Message Digest Algorithm :-

This concept is similar to hash function. Message digest also the fingerprint of the given message.

We take the original msg block of the variable length. Then we use the message digest Algorithm (MD5, SHA) to produce the fixed & small size message digest.



Message Digest Algorithm

Example :- 7391743

Here we use the concept that multiply each digit in the no. with the next digit (exclude 0) & discarding the least digit of the multiplication operation.

Original no. is 7391713

Operation	Result
Multiply 7 by 3	21
Discard first digit	1
Multiply 1 by 9	9
Multiply 9 by 1	9
Multiply 9 by 7	63
Discard first digit	3
Multiply 3 by 4	12
Discard first digit	2
Multiply 2 by 3	6

So, final message digit is 6

Message digest calculation process

MD5 Message Digest Algorithm :-

MD5 is msg digest algorithm that was developed by Ron Rivest. MD5 is a improved and the latest version.

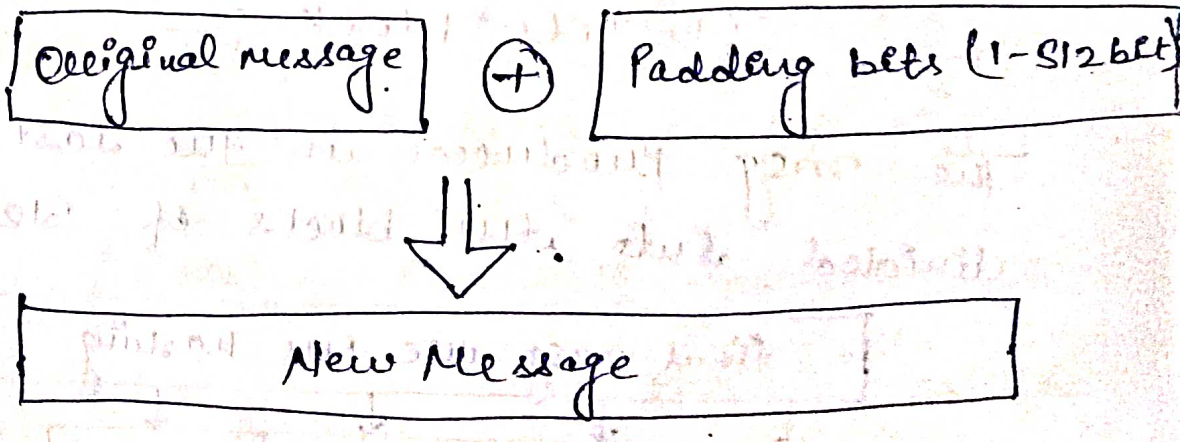
The original message digest algorithm was called as MD1.

MD5 is a fast algorithm that produce 128 bit long msg digest. It take the input block 512 bits as input & produce the 128 bit block of msg digest as output.

Working Of MD5 :- The working of the MD5 msg digest algorithm can be classified into the following step.

Step 1 Padding :- The first step of the MD5 is to add padding bits to the original message. The reason behind this padding is that we want to make the length of the original message equal to the values, that is 64 bit less than the exact multiple of 512 bits. This 64 bits less value will be used for length Padding.

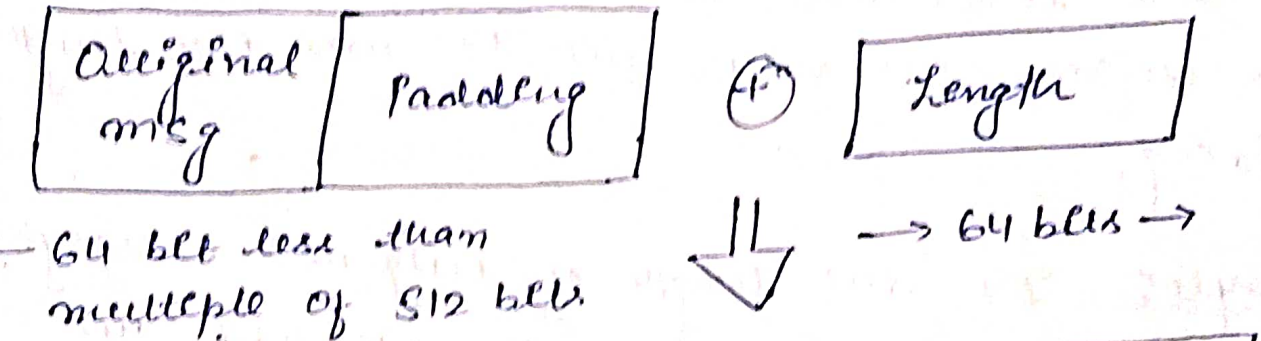
Example :- If the length of original message is 1200 bits, then the we add the Padding of 272 bits to make the msg 1472 bit long. 1472 is exact multiple of 512. (1536 = 512 X 3)



Step: 2 Append the Length Field :-

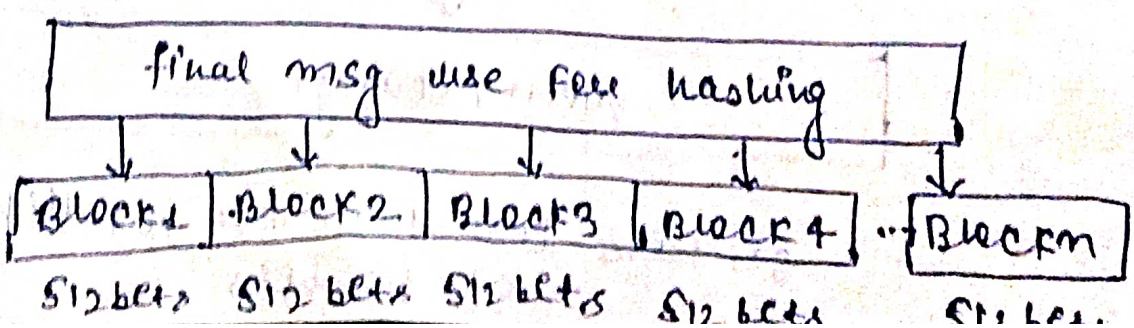
We calculate the length of the original message (i.e. excluding the padding bits). This length is added at the last after the padding bits in the original message.

The length of original msg is expressed in the 64 bits value.



Step 3 :- Divide the Input Final Msg into 512 bits Block :-

The msg produced in the last step is divided into the blocks of 512 bits each.



Step 4 :- Initialize Chaining Variable :-

We take the 4 variables known as chaining variables. They are identified as A, B, C & D. Each of these is a 32 bit number. The Hexadecimal values of these chaining variable.

32bit	A	01	23	45	67
32bit	B	89	AB	CD	EF
32bit	C	FE	DC	BA	98
32bit	D	76	54	32	10

* We starts from 0–9 in the pairs, then A–f in the pair.
* After filling till F, reverse the process in pairs.

A = 0x 01234567
B = 0x 89ABCDEF
C = 0x FEDCBA98
D = 0x 76543210

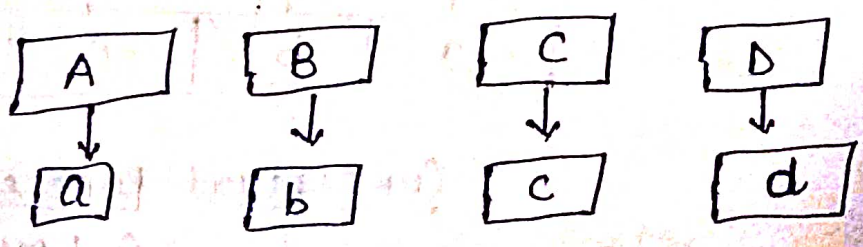
Step 5 :- Process Blocks :- This is the main step of the algorithm.

The working of this step can be classified into the further sub-steps.

Step 5.1

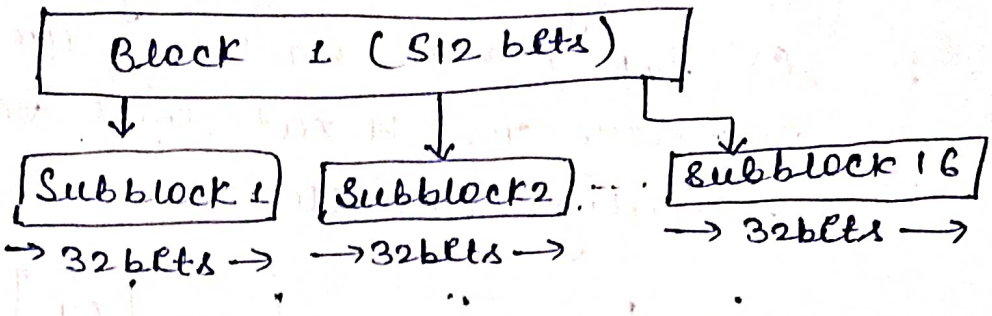
Copy the 4 chaining variable into 4 corresponding variable a, b, c, d.

a = A
b = B
C = C
d = D



Copying chaining variables into temporary variable

Step S2 :- Divide the current 512 bits block into 16 sub-blocks of 32 bits each.



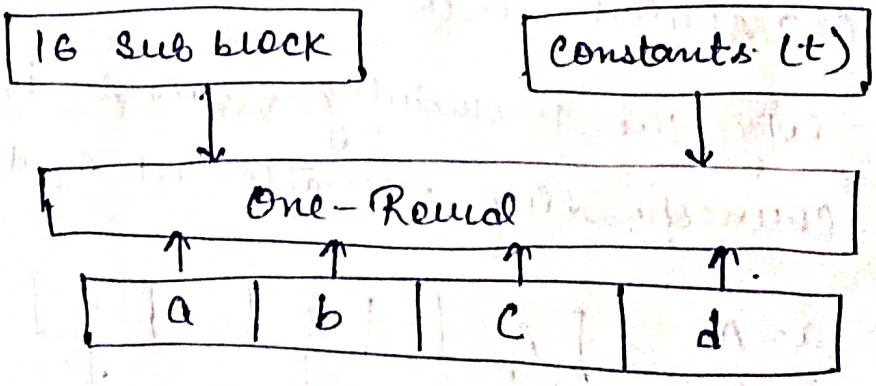
Sub blocks with in a block

Step S.3 :- We have 4 rounds in each round we process all the 16 sub blocks. The input to the each round use the three thing.

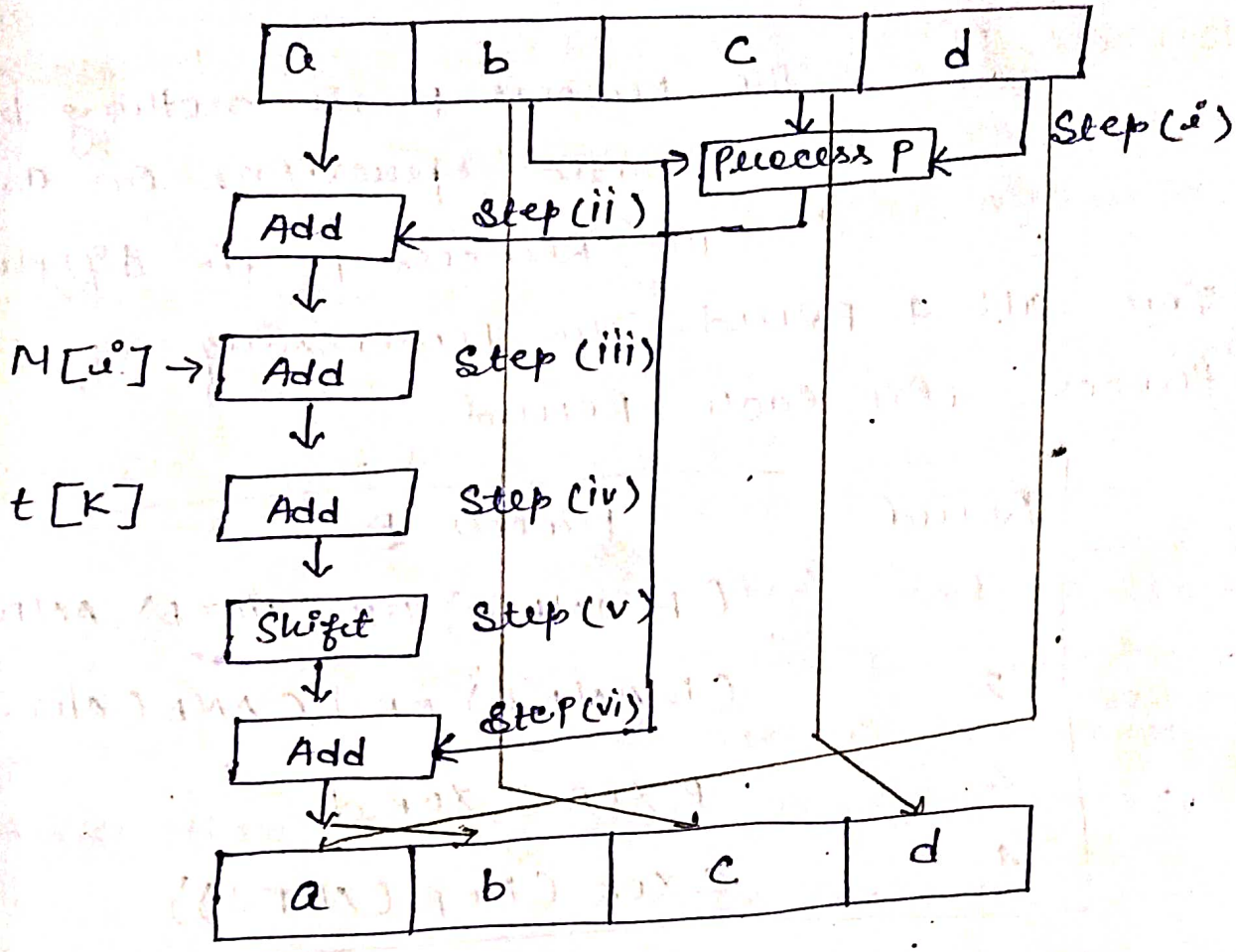
(i) All the 16 bits sub-blocks.

(ii) The variable a, b, c & d.

(iii) Some constants known as t.



Conceptual process with in a Round



One MDS operation

We can mathematically express a single MDS operation:-

$$a = b + (a + \text{Process } P(b,c,d) + M[i] + t[k])^{<<<s}$$

where

a, b, c, d → chaining variables.

Process P → A non-linear operation

M[i] → i^{th} 32 bit word, sub blocks

t[K] → a array of constant

<<< → circular left shift by s bits.

The Process P :-

The Process P is nothing but Some basic operations on a, b, c, d. The Process P is different for all 4 Round. The processing of P Process in each Round.

$F(b,c,d) = (b \wedge c) \vee (b' \wedge d)$
$G(b,c,d) = (b \wedge d) \vee (c \wedge d')$
$H(b,c,d) = b \oplus c \oplus d$
$I(b,c,d) = c \oplus (b \vee d')$

Round	Process P
1	(b AND c) OR (NOT b) AND (d)
2	(b AND d) OR (C AND (NOT d))
3	B XOR C XOR d.
4	C XOR (b OR (NOT d))

Process of P in each Round

The Strength of MD5 :-

The attempt of Rivest was to add as much of complexity & randomness as possible to the MD5 algorithm, so that no two msg digests produced by MD5 on any two different message are equal.

Secure Hash Algorithm

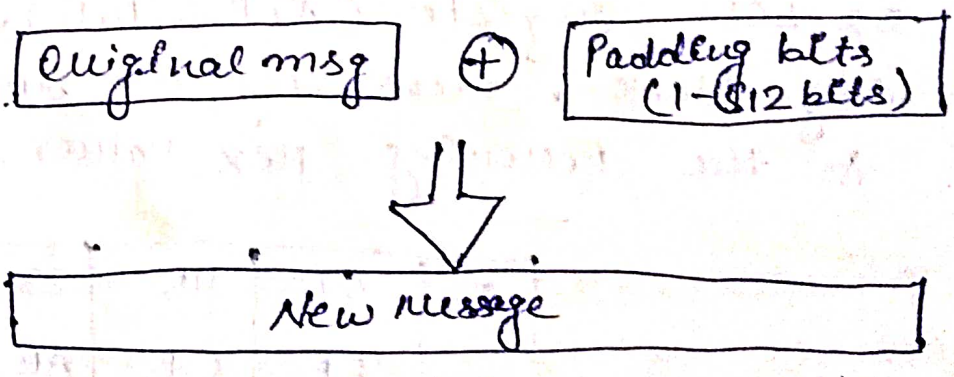
The Secure Hash Algorithm - 1 (SHA-1) is also a one way hash function algorithm used to create digital signatures.

SHA-1 is similar to the MD5 algorithms developed by Ron. Rivest. SHA-1 is slightly slower than MD5, but it is found to be more secure.

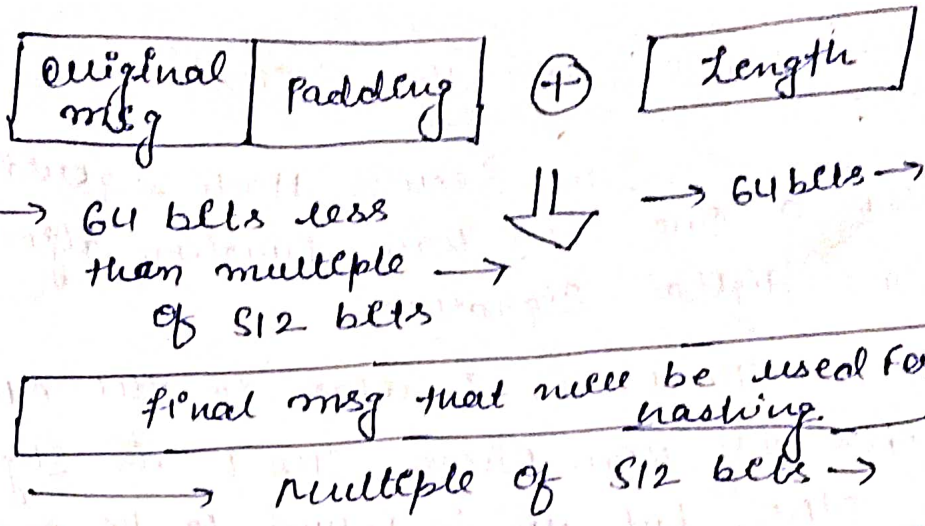
SHA works with any input msg that is less than 2^{64} bits in length. The output of SHA is a msg digest, which is 160 bits in length (32 bits more than the msg digest produced by MD5)

Working of SHA-1 :- Like MD5, the first step

1) **Padding:-** in SHA-1 is to add Padding to the end of the original message. The Padding bit always needed, even if the length of the msg already multiple of 512 bits.

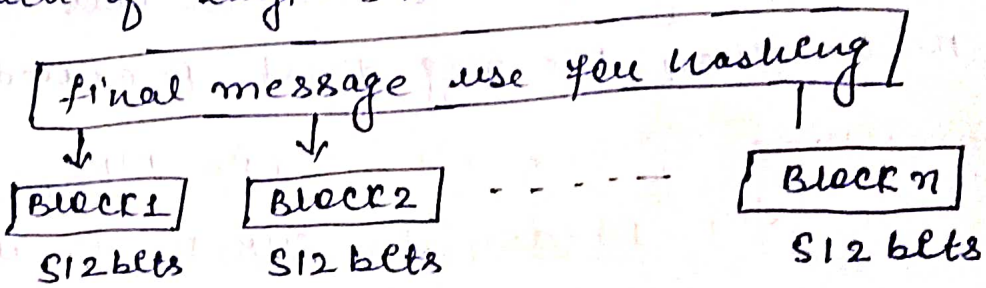


Step 2 :- Append length :- The length of the msg Excluding the length of the Padding is now calculated and appended to the end of the padding as 64 bit-blocks



Step 3:- Divide the Input into 512 bit blocks:-

The input message is now divided into blocks, each of length 512 bits.



Step 4:- Initialize 160-bits chaining variable:-

. The 160-bit buffer consists of five 32-bits registers (A, B, C, D, E). Before processing any block, these registers are initialized to the following Hex values.

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

Step-5 :- Process Block :- Now actual algorithm begin.

Here also, the steps are quite similar to those in MD5.

(i) copy of chaining variable A-E into variable a-e. The combination of a-e, called as abcde will be considered as a single register for storing the temporary final result.

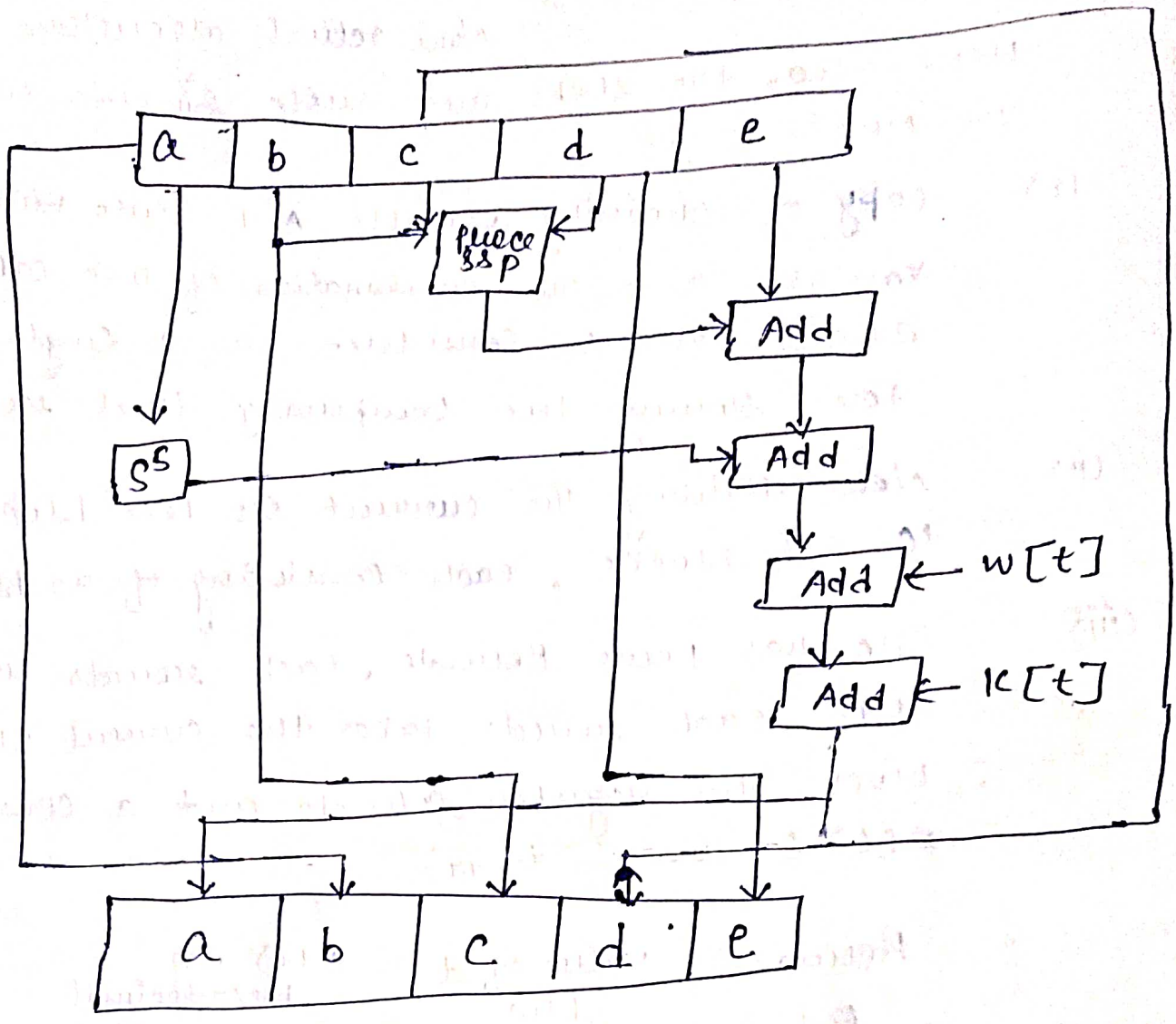
(ii) Now dividing the current 512 bits block into 16 sub-blocks, each consisting of 32 bits.

(iii) SHA has Four Rounds, each rounds has 20 steps. each rounds takes the current 512 bits block, the register abcde and a constant K(t) (where t=0 to 79).

Rounds	value of t	K(t) in hexadecimal
1	blw 0 to 19	5A 82 79 99
2	20 to 39	6E D9 EB A1
3	40 to 59	9f 1B BC DC
4	60 to 79	CA 62 C1 D6

(iv) 4 rounds are used in SHA. 20 iteration in each rounds. So the total 80 iteration in SHA.

The logical operation of SHA :-



Mathematically we can write the iteration process :-

$$abcde = (e + \text{Process } P + S^5(a) + w[t] + K(t))$$

abcde = The register to store the intermediate and final values.

Process P = The logical operation

S^5 = Circular left shift of the 32 bit Sub-blocks by t bits

w(t) = A 32 bits value

K(t) = One of the 4 additive constants

The Process P :-

The Process P of SHA algorithm has different complex mathematical formula in each 4 rounds.

Round	Process P
1	(b AND c) OR (NOT b) AND (d))
2	b XOR c XOR d
3	(b AND c) OR (b AND d) OR (c AND d)
4	b XOR c XOR d

Security of SHA-1 :-

SHA-1 is the most secure message digest algorithm.

There have been no successfull attacks reported against the SHA. Because it use 80 iterations and produces 160-bits message digest.

Comparison between MD5 and SHA-1 :-

Comparison	MD5	SHA
Security	less secure than SHA.	Higher secure than MD5.
Msg digest length	128 bits	160 bits
original message	2^{128} operations	2^{160} operation
message digest	2^{64} operation required to break	2^{80} operations to break
Speed	only 64 iteration	only 80 iteration

Digital Signature :-

Digital Signatures are public-key primitives of message authentication.

It is a technique that binds a person / entity to the digital data. This binding can be independently verified by receiver as well as any third party.

Digital Signature is a cryptographic value that is calculated from the data.

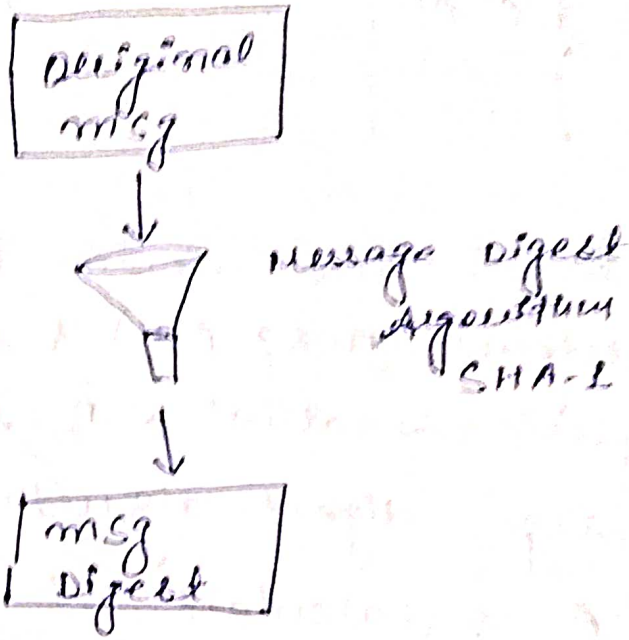
Digital Signature Standard makes use of SHA-1 algorithm for calculating the msg digest over an original message & uses the message digest to perform the digital signature. DSS makes use of an algorithm, called as Digital Signature Algorithm. DSS is Standard & DSA is actual algorithm.

RSA & Digital Signatures :- We have mentioned that RSA can be used for performing digital Signature.

Let us understand how this works in step by step fashion.

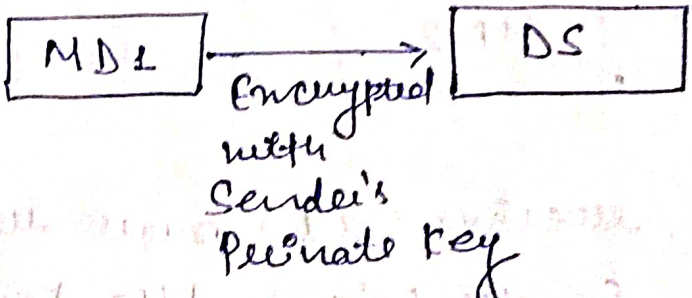
SENDER (A) wants to sends a msg (M) to receiver (B) along with the digital Signature (S) calculated over message (M)

Step 1:- The sender (A) uses the SHA-1 msg digest algorithm to calculate the message digest (MD1) over the original message (M).

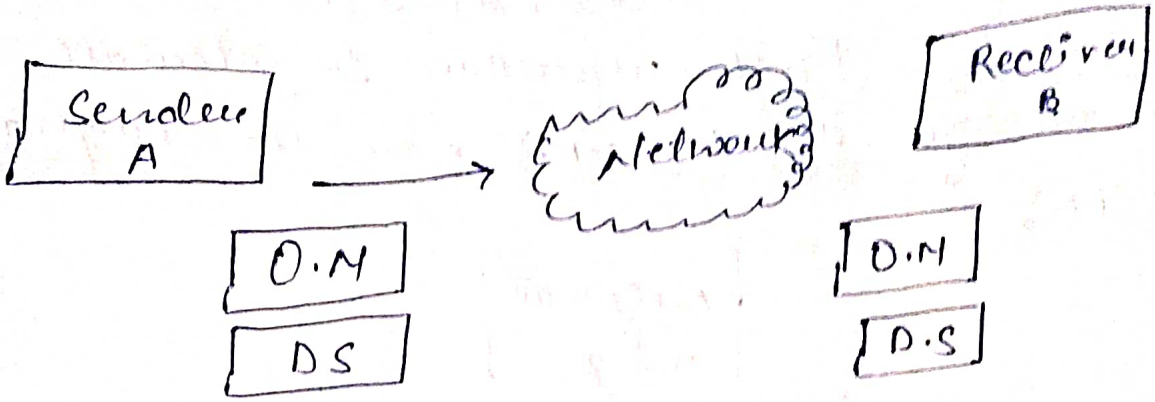


Message digest calculation

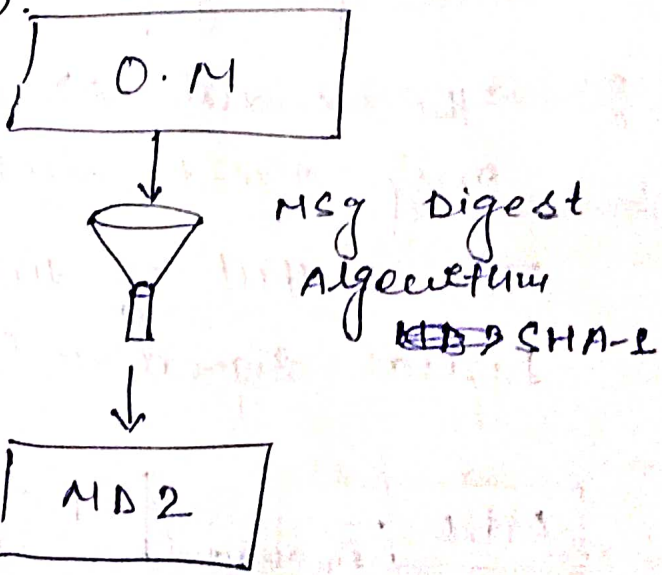
Step 2:- The sender (A) now encrypts the msg digest with her private key. The output of this process is called Digital Signature (DS)



Step 3:- Now Sender's (A) sends the original msg (M) along with DS to the receiver (B)



Step 1 :- After the receiver (B) receives the original msg (M) & sender's (A) digital signature, B uses the same message digest algorithm as was used by A & calculate it own msg digest (MD2).



Step-5 :- The receiver (B) now uses the Sender (A's) Public key to decrypt the digital signature.

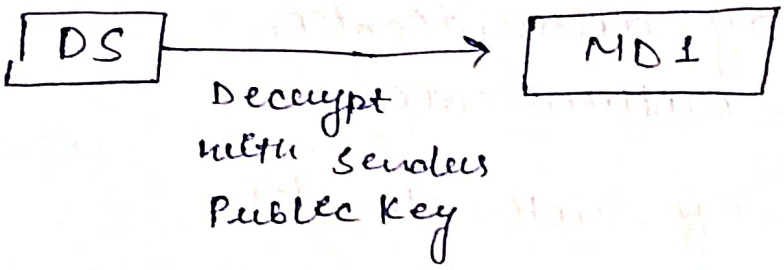
A had used her Private key to encrypt the O.M & also use the A's Public

key to decrypt the message.

Step 6 :- B now compares the following 2 message digests :-

•) MD2, which is calculated in step 4

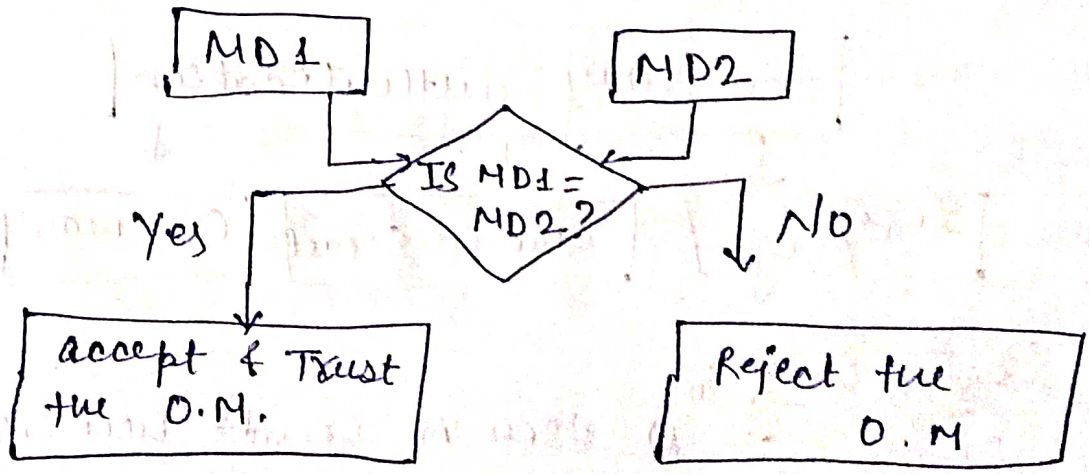
•) MD1, which is retrieved from A's digital signature in step 5.



if MD1 = MD2

1) B accept the original message (M) as the correct, unaltered message from A.

2) B is also reject the msg, if msg are not same, altered.



Digital signature verification

Authentication Protocol :-

In this protocol we trying to authentication of each other. Both the parties are use authentication for the secure communication.

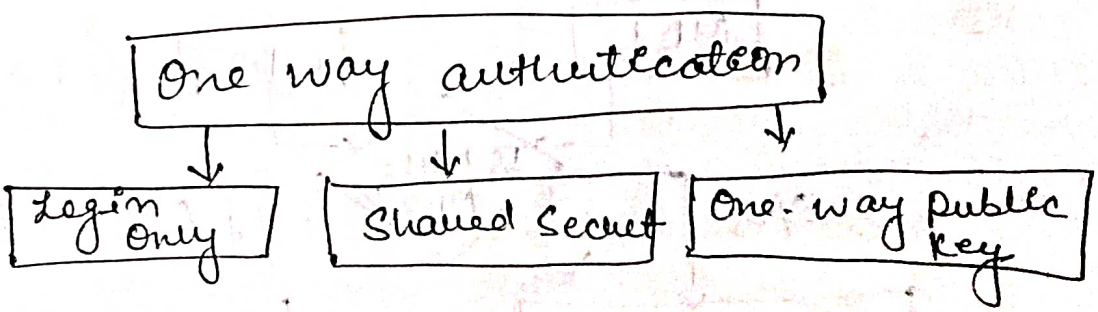
There are two broad level schemes for carrying out the outlined.

→ One way authentication

→ mutual authentication

(i) One way authentication :-

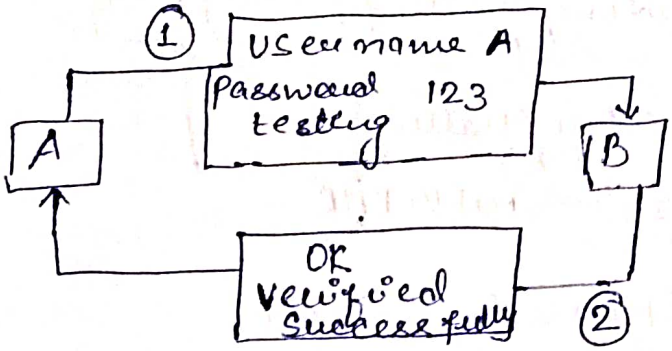
The idea behind one-way authentication is simple if there are two users A & B. B authenticates A, but A does not authenticate B. we call it one-way authentication. There are various ways in which type of authentication scheme can be implemented.



(i) Login Only :-

1) User A sends her user name & password in the plain text form to the other user B.

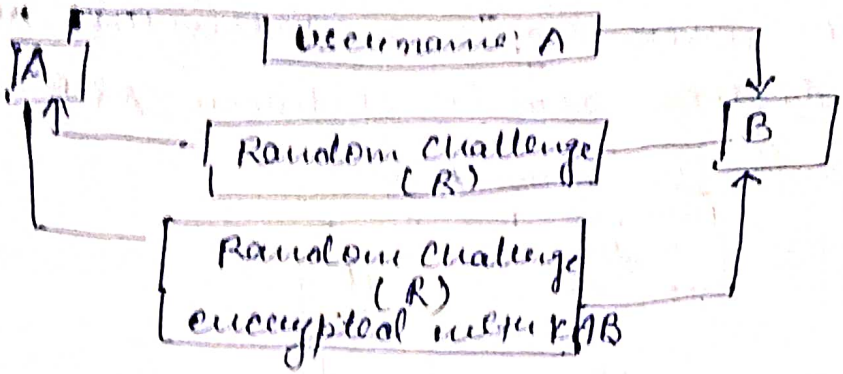
2) B verifies the user name and password. if the user name & password are correct, communication starts between A & B.



Login only authentication

Shared secret :- There is an assumption that A & B have agreed on a shared Symmetric key KAB.

1. A sends her user name and password to B
2. B creates a random challenge R and sends it to A.
3. A encrypt the random challenge (R) with the shared symmetric key b/w A & B (KAB) and sends the encrypted R to B. B also encrypts the original random challenge (R) with the same shared symmetric key (KAB). If this encrypted challenge matches with the one sent by A, B considers A to be authentic

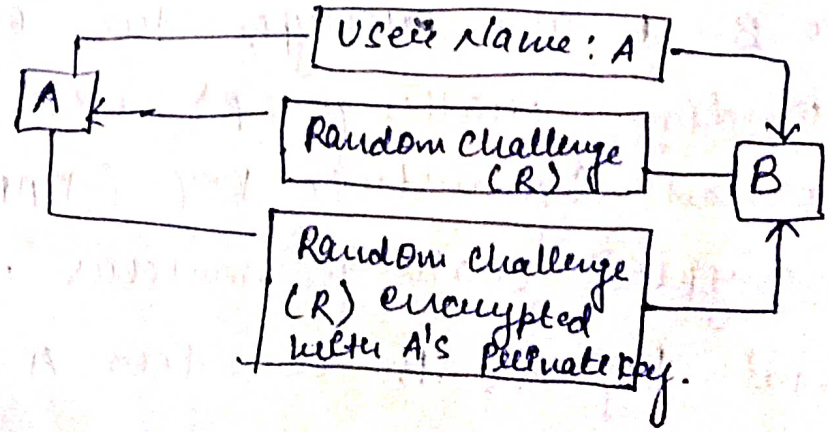


Shared Secret

One-way Public Key:-

The idea is simple behind this

1) A sends her user name to B
2) B sends the random challenge (R) to A.
3) A encrypt the random challenge (R) with her private key & send it to B. B uses the Public key of A to decrypt the encrypted random challenge, match with the original Random Challenge (R).

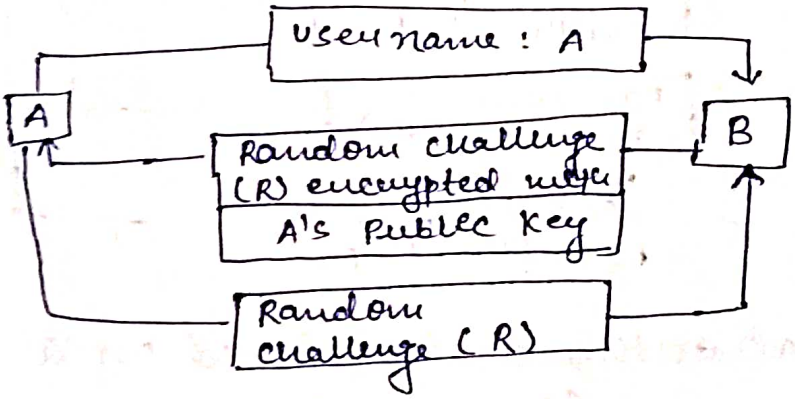


One way Public key - Approach-1

1) A sends her user name to B.

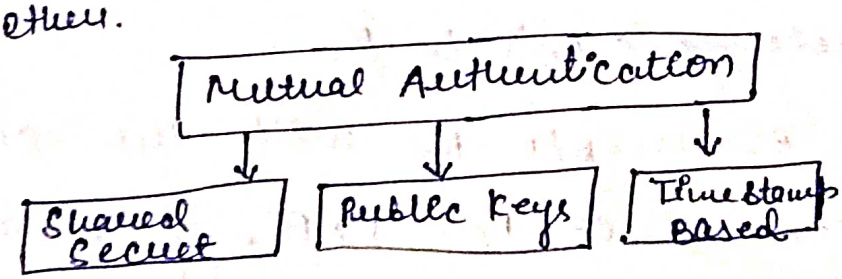
2) B creates the random challenge (R) and encrypts it with the Public Key of A, B sends this encrypted random challenge to A.

3) A decrypted the encrypted random challenge (R) with the private key & send it to B. B matches it with the original Random challenge (R)



Mutual Authentication :-

In this both A & B both authentecate each other.



1) Shared secret :- This protocol assumes that A & B have a shared symmetric key K_{AB}.

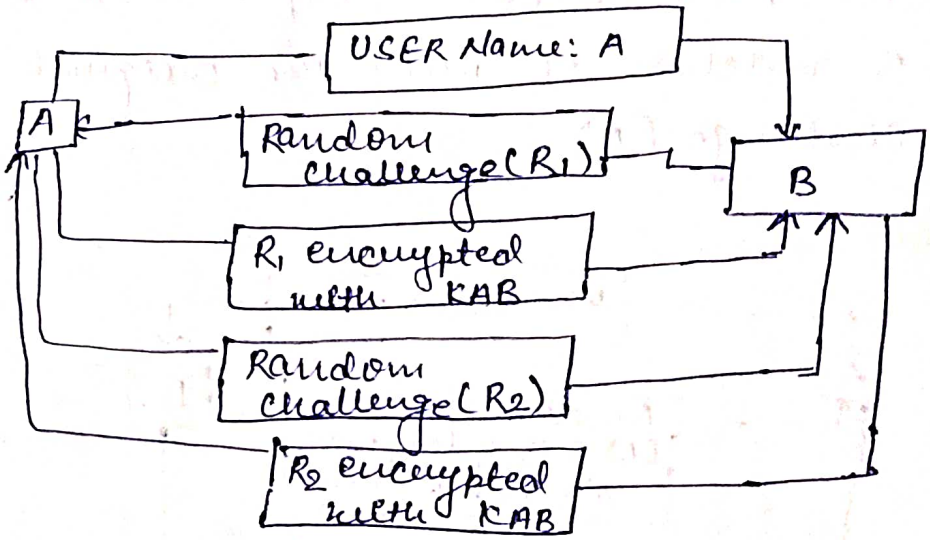
1) A Sends her user name to B.

2) B Sends a random challenge R_1 to A.

3.) A encrypts R_1 with KAB and sends it to B.

4) A sends a different random challenge R_2 to B.

5) B encrypts R_2 with KAB & sends it to A.

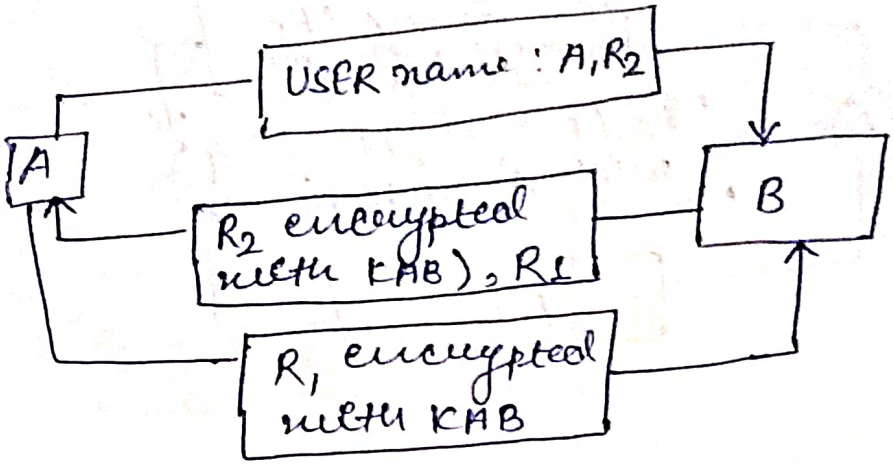


Mutual authentication based on a Shared Secret

1) A sends the user name and a random challenge (R_2) to B.

2) B encrypts R_2 with the shared Symmetric key KAB, generated a new random challenge (R_1) and send these two to A.

3) A verifies R_2, encrypts R_1 with the shared Symmetric key KAB, & send it to B. B verifies R_1.

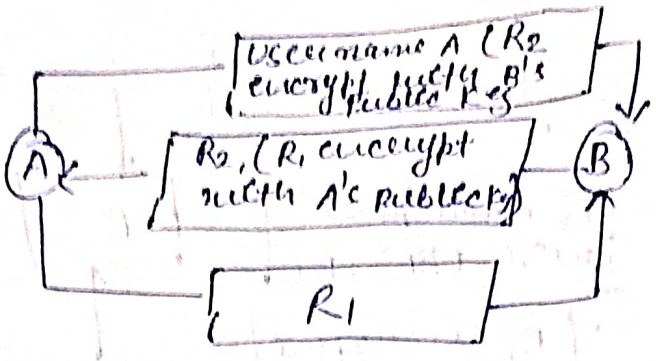


Optimized mutual authentication based on a shared secret

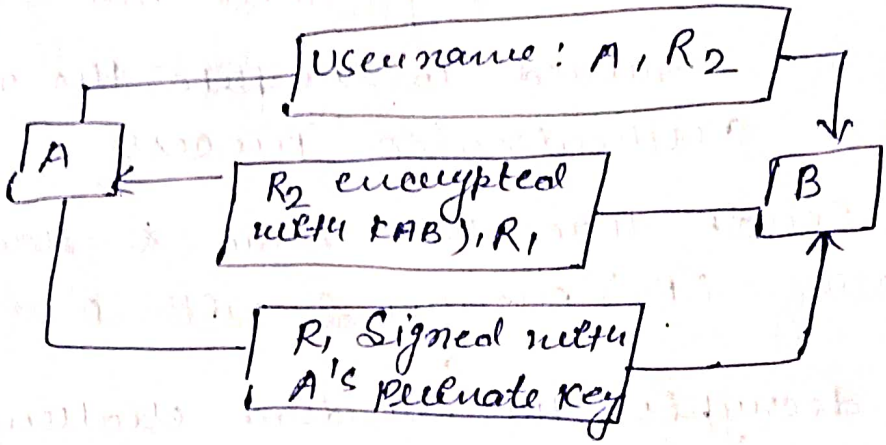
ublic keys :- If A & B know each other's Public key, three message are required to complete the mutual authentication process.

1. A sends her user name & random challenge (R2) encrypted with B's Public key.
2. B decrypts the random challenge (R2) with his private key. B Creates a new random challenge (R1) and encrypt it with A's Public keys. B Send two thing (decrypted R_2 & encrypted R_1) to A.
3. A decrypt the random challenge (R_1) with her private key and send it's to B.

B verifies R_1.



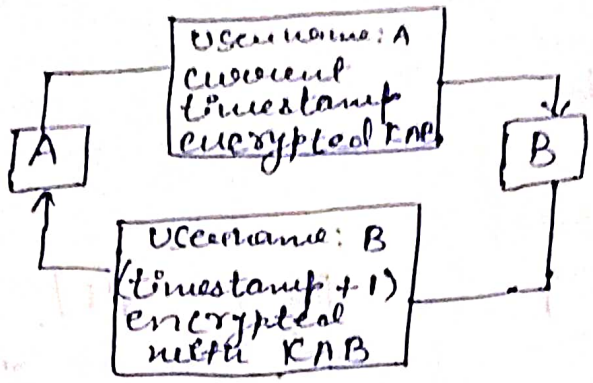
1) A sends her user name and R_2 to B.
2.) B encrypt R_2 with his Private key & send it & R_1 to A.
3) A Signs R_1 and returns it back to A.



Time-Stamp :- We can reduce the mutual authentication process to just two steps by using timestamps.

1> A Sends her user name & current timestamp encrypted with a Shared Symmetric key (KAB) to B.

2.) B retrieves the timestamp by decrypting using KAB and One to the timestamp.
B encrypts the result with KBA (not KAB!) and send it to A, along with her user name.



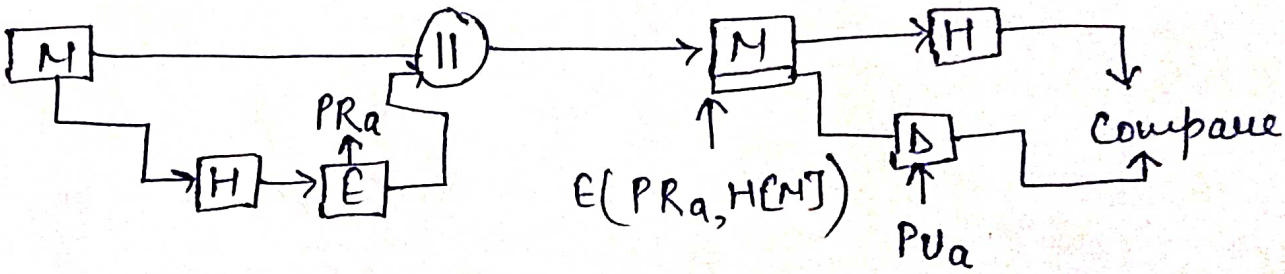
Mutual Authentication using time stamps

DSS Standard :- Digital Signature Standard

DSS makes use of the Secure Hash algorithm & presents new digital signature techniques, the digital signature algorithm.

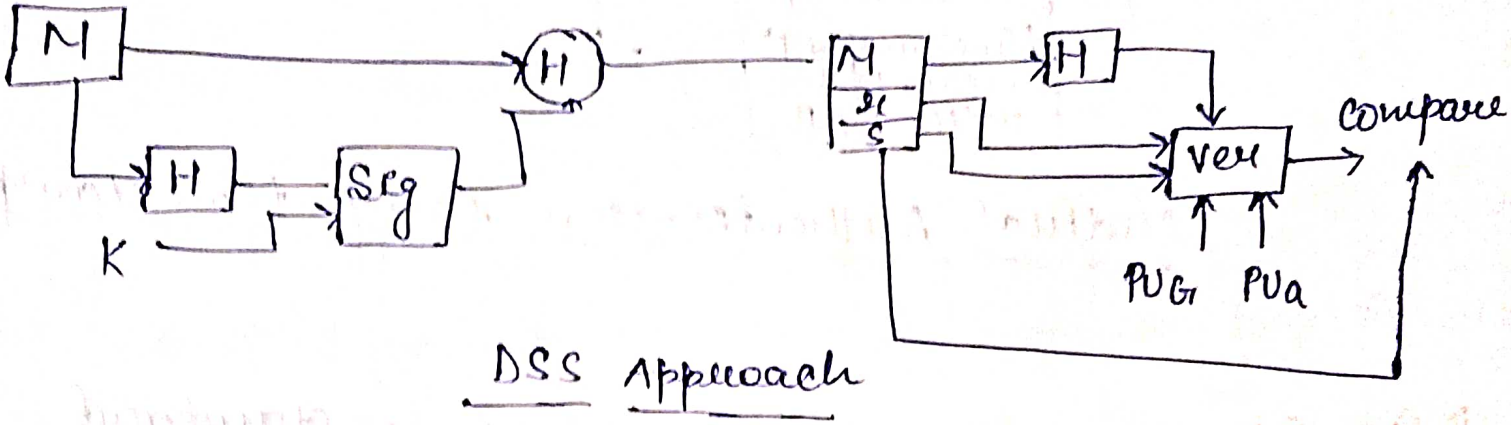
The DSS Approach :- The DSS uses an algorithm that is designed to provide only the digital signature function.

RSA Approach is use for this Digital Signature Standard. It is a Public-Key techniques.



a) RSA approach

b) DSS approach



DSS Approach

IDEA :- International Data Encryption Algorithm :-

IDEA is a symmetric block cipher algorithm.

It was meant to be a replacement for the Data Encryption Standard.

1) Here plain text in 64 bit.
2) Key is of 128 bit. And it is divided in 52 Sub keys.
3) Cipher text is same as the plain text that is 64 bit.
4) Number of identical rounds are 8 where in each round 6 key are used.
5) Like this 48 keys in last round another 4 keys (6 * 8 = 48 + 4 = 52 total) are being used in both the Encryption and decryption process.
6) Which different operations are used.
   - → XOR
   - → Addition
   - → Multiplication

→ The 128 bit key is divided into 8 Sub Parts that is 16 bits each.

→ K_1 to K_{52} keys are generated.

Sequence of operation in One Round :-

1) Multiply P_1 and K_1
2) Add P_2 and second K_2
3) Add P_3 and third K_3
4) Multiply P_4 and K_4
5) Step (1) ⊕ Step 3
6) Step 2 ⊕ Step 4
7) Multiply Step 5 with K_5.
8) Add result of step 6 and step 7.
9) Multiply result of Step 8 with K_6
10) Add result of Step 7 and step 9
11) XOR result of steps 1 and step 9
12) XOR result of Steps 3 and Step 9.
13) XOR result of Steps 2 and Step 10,
14) XOR result of Steps 4 and Step 10.

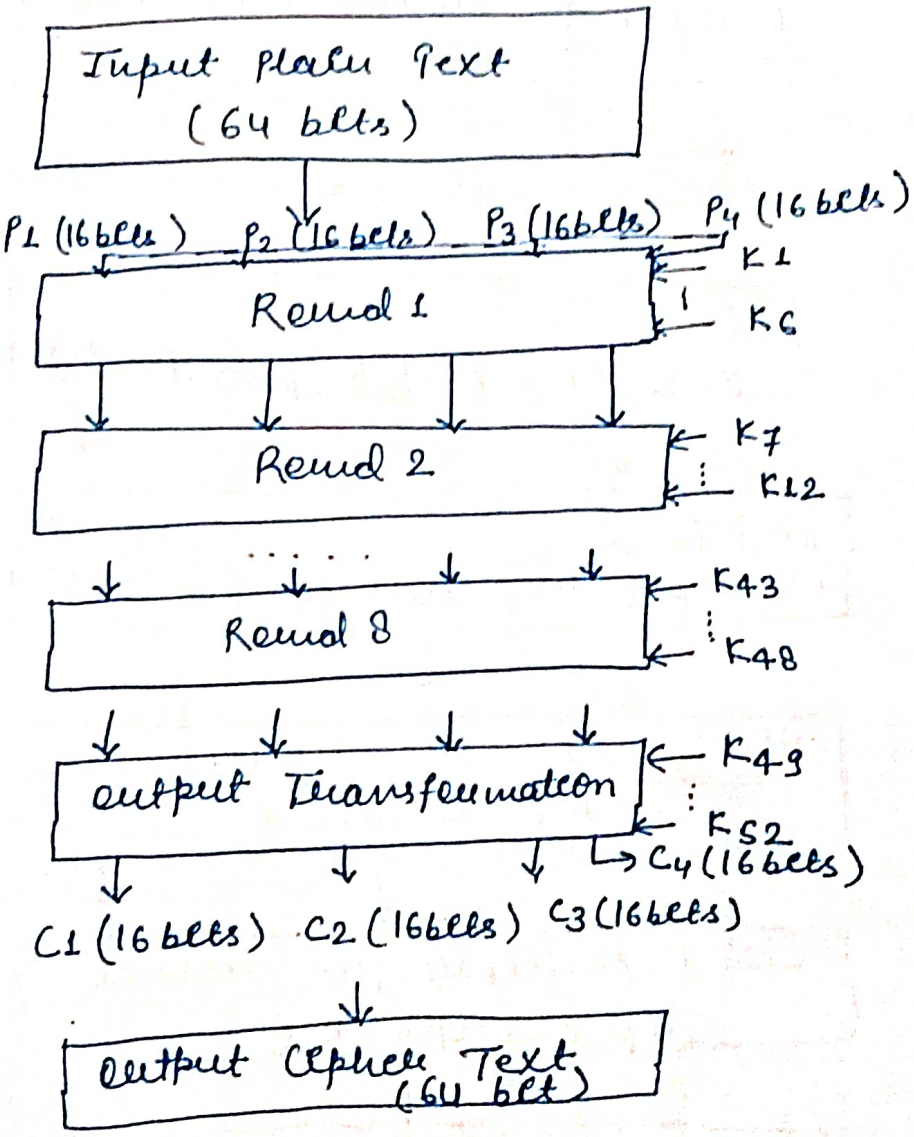
Same operations are performed in 8 rounds--

1) Multiply P_1 with K_{43}

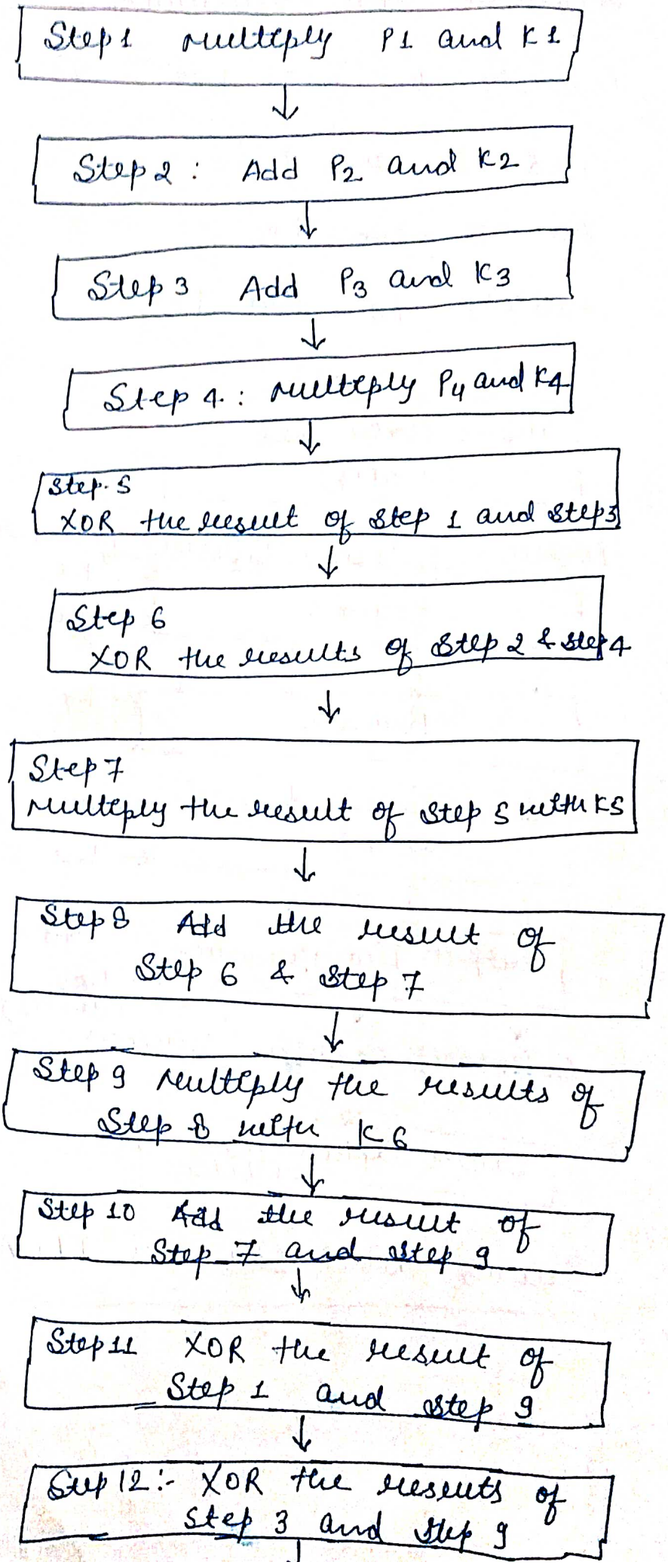
2) Add P_2 and K_{50}

3) Add P_3 and K_{51}

4) Multiply P_4 and K_{52}.



Broad level Steps in IDEA


Step 1 multiply P1 and K1
Step 2 : Add P2 and K2
Step 3 Add P3 and K3
Step 4. : multiply P4 and K4
Step 5
XOR the result of step 1 and step3
Step 6
XOR the results of Step 2 & step 4
Step 7
multiply the result of Step 5 with K5
Step 8 Add the result of
Step 6 & Step 7
Step 9 multiply the results of
Step 8 with K6
Step 10 Add the result of
Step 7 and step 9
Step 11 XOR the result of
Step 1 and step 9
Step 12 :- XOR the results of
Step 3 and Step 9

Step 13:
XOR the result of step 2 and step 10

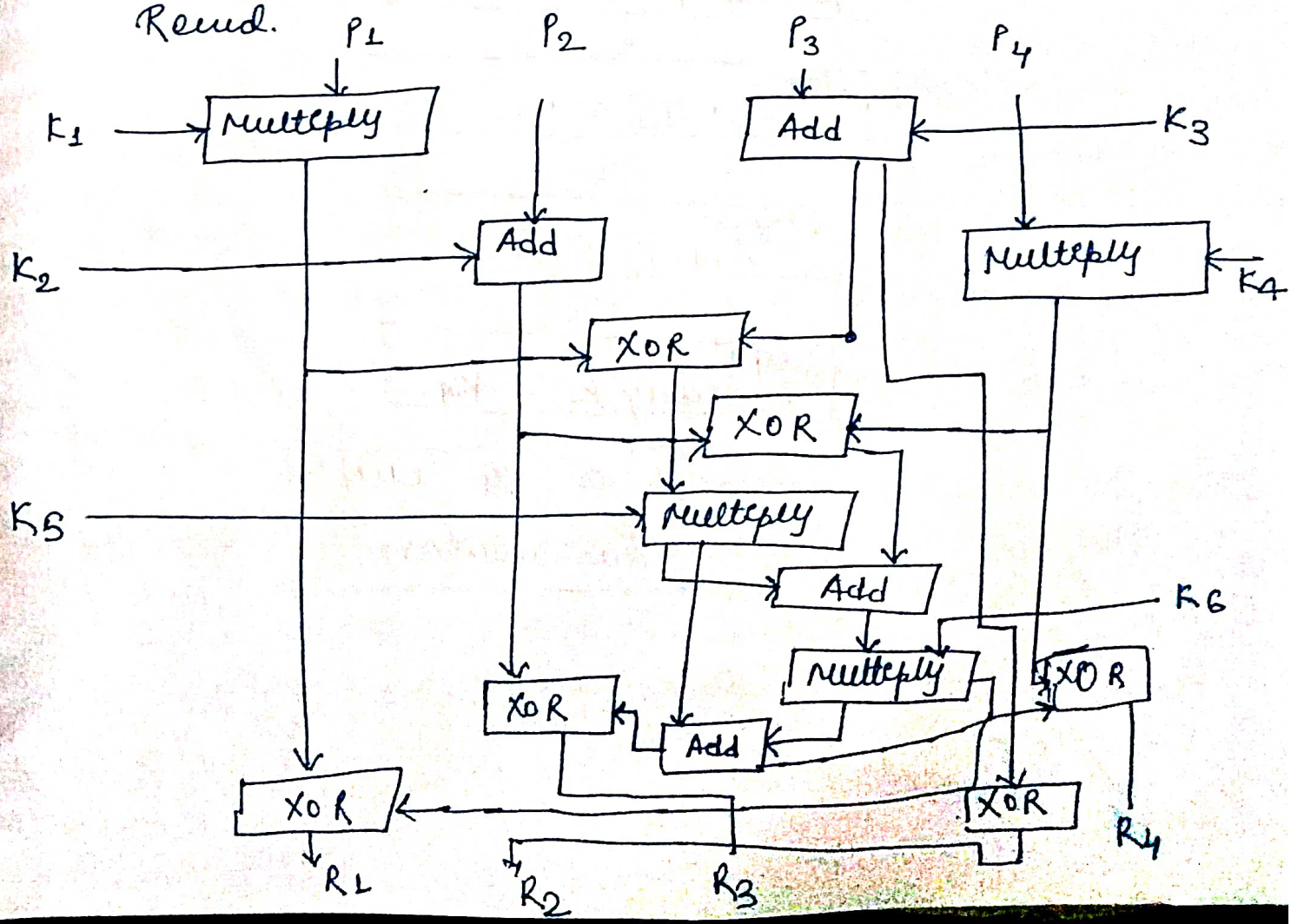
↓

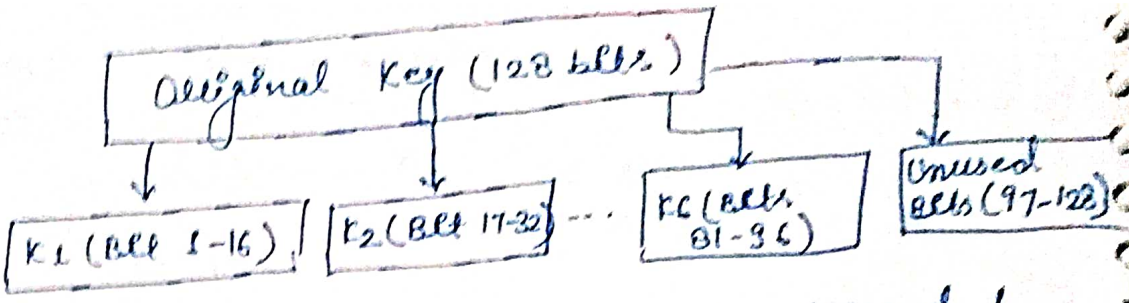
Step 14:-
XOR the result of step 4 & step 12

Detail of one round in IDEA

Sub-key Generation for a Round :-

First Round :- The initial key consists of 128 bits, from which 6 sub keys K_1 and K_6 are generated for the first Round. Since K_1 and K_6 consists of 16 bits each, out of the original 128 bits, the first 96 bits are used for the first Round.





Sub-key generation for round 1

Output Transformation :-

The output Transmation is one time operation. It takes place at the end of the 8th Round. 64 bit value divided into 4 sub-blocks (say R_1 to R_4, each consisting of 16 bits). Four sub-keys are applied here and not 8 times.

Step 1
Multiply R_1 and K_1

↓

Step 2
Add R_2 & K_2

↓

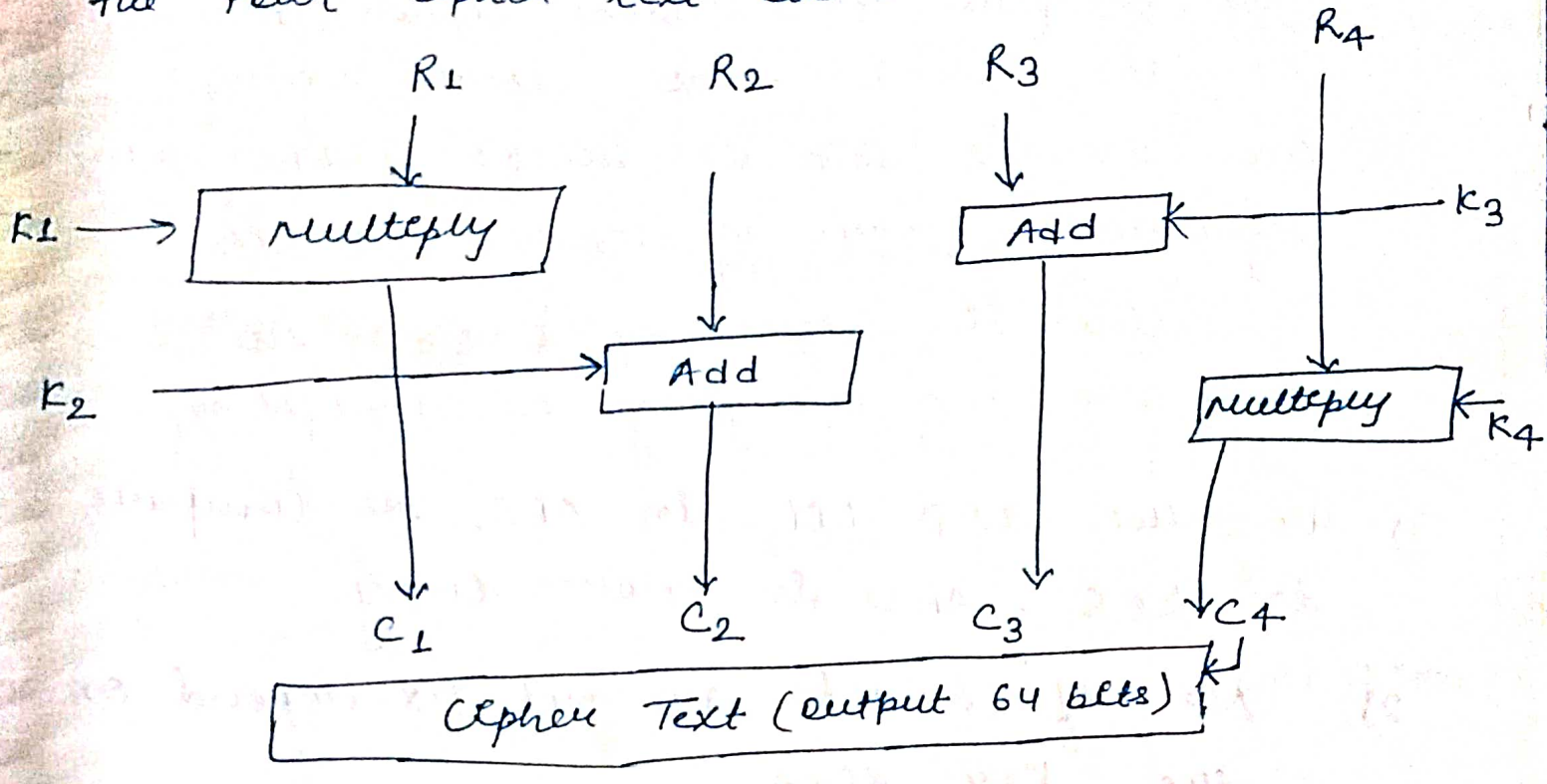
Step 3:-
Add R_3 & K_3

↓

Step 4:-
Multiply R_4 & K_4

Details of the output Transmation

The output of this process is the final 64-bit cipher text. which is the combination of the four cipher text sub-blocks.



Output Transformation Process

IDEA Decryption :- The decryption process is exactly the same as the encryption process. There are some alteration in the generation and pattern of sub-keys. The decryption sub-keys are actually inverse of the encryption sub-keys.

The Strength Of IDEA :- IDEA uses a 128-bit key, which is double than the key size DES. To Break Into IDEA 2^{128} encryption would be required.