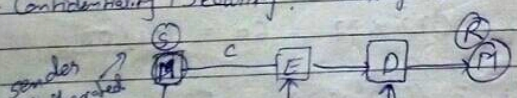
⇒ Sessional - 2

Public key cryptography :

Public (stored in public key register)

Private - present to the end users

- Authentication - source of message should be Authentic
- Confidentiality / Security - message should not be reveal inbetween



⑤ M → [E] [D] → M ⑧ ensure confidentiality

Rpu Rpr

ensure
⇒ Authentication & Confidentiality both

Sender M → [E] × [E] Y [D] × [D] → M

Prs PuR PrR Pus

$E_{PuR}(E_{Prs}(M))$

→ Application of public key cryptography
- producing digital signature (Authentication)
- encryption - dec
- key extension.

7/8

⁂ RSA Algorithm. (Rivest - Shamir - Adleman)

- used in multiple ways
  - Encry / Decpt
  - Digital signature
  - key exchang / generation.

→ $C = M^e \bmod n$

$M = C^d \bmod n$

$\phi(n) = (p-1)(q-1) = 160$

$p = 17$

$q = 11$

$n = 187$

$ed \bmod n = 1$

$1 < e \& d < 160$

$\therefore e = 7$ $\quad 7 \times d \bmod 160 = 1$

$d = 23$

$(\because 161 \bmod 160 = 1)$

⇒ $M = 88$, $PU = [7, 187]$, $PR = [23, 187]$

$C = 88^7 \bmod 187 = 11$

$M = 11^{23} \bmod 187 = 88$

⇒

① Identify p, q → prime no.

② $n = p * q$

③ $\phi(n) = (p-1) * (q-1)$

④ Integer e, such that $\gcd(e, \phi(n)) = 1$ (public key / private key chosen number) $\quad 0 < e < \phi(n)$

public key = {e, n}

(5) Calculate private key d

$ed \bmod (\phi(n)) \equiv 1$

private key = {d, n}

ex (1) p = 17, q = 11

(2) n = 187

(3) $\phi(n) = 160$

(4) e = 7 ∵ gcd(e, $\phi(n)$) = 1

(5) $d = e^{-1} \bmod \phi(n)$

e, d < 160

7, d < 160

$7 \cdot 23 = 160 + 1 \Rightarrow 23 * 7 > 160$

$160 * 16 + 1$

ignore this (Following euclids theorem)

⇒ Euclid Theorem :

1. A ← a , B ← b
2. if B = 0
   Return A = gcd(a, b)
3. R = A mod B
4. A ← B
5. B ← R
6. goto step 2.

greatest common divisor

gcd (1970, 1066) = 2.

$1970 = 1 \times \underbrace{1066}_{A} + \underbrace{904}_{B}$ gcd (1066, 904)

$1066 = 1 \times 904 + 162$ gcd (904, 162)

$904 = 5 \times 162 + 94$ gcd (162, 94)

$162 = 1 \times 94 + 68$ gcd (94, 68)

$94 = 1 \times 68 + 26$ gcd (68, 26)

$68 = 2 \times 26 + 16$ gcd (26, 16)

$26 = 1 \times 16 + 10$ gcd (16, 10)

$16 = 1 \times 10 + 6$ gcd (10, 6)

$10 = 1 \times 6 + 4$ gcd (6, 4)

$6 = 1 \times 4 + 2$ gcd (4, 2)

$4 = 2 \times 2 + 0$ gcd (2, 0)

~~$2 = 1 \times 2 + 0$~~

common factor

$23 = 1 \times 23$
$15 = 1 \times 3 \times 5$ } relative prime with each other.

$\Rightarrow$ (7, 160) , d = 1 (From euclid Theorem)

$160 = 22 \times 7 + 6$ gcd (7, 6)

$7 = 1 \times 6 + 1$ (6, 1)

gcd = 1

Ans: $e^{-1} \bmod \phi(n)$.

12/9 RSA

$\Rightarrow$ $p = 7, q = 11, n = 77, \phi = 60$

$\Rightarrow$ e $60 = 5 \times 3 \times 2 \times 2$

3 7 11 13 19 . .

Let, $e = 13$

$d = e^{-1} \bmod \phi(n)$

$= 13^{-1} \bmod \phi(n)$.

↓

extended euclidean algo.

$ax + by = \gcd(a,b) = \gcd(60,13) = 1$

$a = 60, \quad b = 13$

	a	b	r	k (quotient)
1	1	0	60 (a)	—
2	0	1	13 (b)	$(60/13) = 4$
3	1	-4	8	$(13/8) = 1$
4	-1	5	5	$8/5 = 1$
5	2	-9	3	$5/3 = 1$
6	-3	14	2	$3/2 = 1$
7	5	-23	1	stopping condition.

$a_3 = a_1 - a_2 * k_2$
$= 1 - 0 * 4 = 1$

$b_3 = b_1 - b_2 * k_2$
$= 0 - 1 * 4 = -4$

$D_3 = D_1 - D_2 * k_2$
$= 60 - 13 * 4 =$

$a_4 = 0 - 1(1) = -1$

$b_4 = 1 - (-4) * 1 = 5$

$D_4 = 13 - 8 * 1 = 5$

$a_5 = 1 - (-1) * 1 = 2$

$\therefore d = b = -23$

$d = e^{-1} \bmod \phi(n)$

$ax + by = 1$

$(60)\,5 + 13 * (-23) = 1$ ✓ satisfied.

⇒ when d is -ve

$d = d + \phi$

$d = -23 + 60$

$\boxed{d = 37}$

⇒ if $d > \phi$

$d = d \bmod \phi$.

⇒ if d is +ve & less then ϕ

$d = d$.

→ public key = $\{e, \phi\} = \{13, 60\}$

private key = $\{d, \phi\} = \{37, 60\}$

⇒ Encryption

M = plain text

$C = M^e \bmod \phi(n)$

$M = e^d \bmod \phi(n)$

M = 10

C =

public key = {e, n}

private {d, n}.

gcd (a, b) = 1

cd (a, b)

(60, 13) = 1

$C = 10^{13} \bmod 77$

$= \left[(10^8 \bmod 77) * (10^4 \bmod 77) * (10 \bmod 77) \right] \bmod 77$

= 10

$(103 * 7) \bmod 561 = 16.$

⇒ Modular exponentiation algo.

```
c ← 0, d ← 1
for i ← k down 0
    do  c ← 2 * c
        d ← d^2 mod n
        if b_i = 1
            then  c ← c + 1
                  d ← (d * a) mod n.
return d.
```

⇒ $a^b \bmod n$ algo.

$(10^{13} \bmod 77 =)$ 560 bi 1000110000

⇒ $7^{560} \bmod 561 = 1$ Ci: 1 2 4 8 17 35 70 140 280 560

$d_2 = 49^2 \bmod 561 = 157$

$d_3 = 157^2 \bmod 561$

d = 7 49

560	bi	1	0	0	0	1	1	0	0	0	0
	Ci	1	2	4	8	17	35	70	140	280	560
	d	7	49	157	526	160	241	298	166	67	1

$10^{13} \bmod 77$ $\quad a=10, b=13$

13 bi 1 1 0 1

ci 1 2 6 13

di 10 76 1 10 (CT)

$100 \times 77 = 23$

$(23*10) \times 77 = 76$

$(76)^2 \times 77$

$(1*10) 77$

encrypt

e.

$(CT) \leftarrow 10^{13} \bmod 77 = 10$

decrypt

d.

$(PT) \leftarrow 10^{37} \bmod 77 = 10$ (PT)

bi 1 0 0 1 0 1

ci 1 2 4 9 18 37

di 10 23 67 76 1 10 (PT)

⇒ $p = 17,\ q = 11,\ e = 7,\ d = ?$

(Plain text) $M = 88$ $\quad a = \emptyset = 160$

(Cipher text) $C = ?$ $\quad b = e = 7$

$$n = p * q = 17 * 11 = 187$$

$$\emptyset(n) = (p-1)(q-1) = 160$$

→ $ax + by = gcd(a,b) = gcd(160, 7) = 1.$

	a	b	D	k (Quotient)
1	1	0	160	—
2	0	1	7	$(160/7) = 22$
3	1	-23	1	$7 - 1 = -7$
4				

$a_3 = 1 - 0 * 23 = 1$

$b_3 = 0 - 1 * 23$

$D_3 = 160 - 7 * 23 = -1$

Q4

=> Sima publishes her RSA key {37, 77}, Bina wants to send secret message M to Sima. Value of M is 2. ~~Both~~ What ciphes Text does Bina sends.

Q5 Calculate 5^{596} mod 1234 using fastes modular expmentiation algorithm. Ans 1013

Ans 4

M = 2, key = {37, 77}

e, n ← public key

$C = M^e \bmod n$

$2^{37} \bmod 77$ n=77, p.1 7 11

bi	1	0	0	1	0	1
ci	1	2	4	9	18	37
di	2	4	16	5	36	51

CT = 51

Ans 5

5^{596} mod 1234

bi	1	0	0	1	0	1	0	0	
ci	1	2	4	9	18	37	74	149	298	596
di	5	25	625	937	565	569	453	541	157	1013

$\phi(n) = 60$ $\quad M = 51^{d} \bmod 77$

d - (extended euclied theorem)

~~d = -23~~

~~d = 37~~

$d = e^{-1} \bmod \phi(n)$

$ax + by = \gcd(a,b) = \gcd(60, 37) = 1$

$a = 60 \quad , b = 37$

	a	b	D	K
1	1	0	60	–
2	0	1	37	$60/37 = 1$
3	1	-1	23	$37/23 = 1$
4	-1	2	14	$23/14 = 1$
5	2	-3	9	$14/9 = 1$
6	-3	5	5	$9/5 = 1$
7	5	-8	4	$5/4 = 1$
8	-8	13	1	
9				
10		d		

$d = 13$

Vishwa

DATE
PAGE NO.

$M = 5^{[illegible]} \bmod 77$

bi	1	0	[illegible]	[illegible]		1	1	0	1
ci	1	2	5	[illegible]		1	3	6	13
di	5	60	[illegible]	[illegible]		5	[illegible]	15	2

19/8

☆ Key distribution

① Public Announcement.
② Publically available directory.
③ Public key authority.
Public key certificates.

⇒ secret key.

20/8 Diffie Hellman Key exchange Algorithm.

Global elements
⇒ q prime
α α<q & it is primitive root of q

$P_A = 3 \qquad 3^a \bmod 13$

2 3 01 2 3 01

Sender & Receiver should know the Global elements
Same secret key will be generated ($K_A = K_B$)

- secrete key calculated independently.. not transferred between sender & receiver

α, q

Privately chosen → (A) select x_A, $x_A < q$

$Y_A = \alpha^{x_A} \bmod q$

(B) select x_B, $x_B < q$

calculate $Y_B = \alpha^{x_B} \bmod q$

Y_A →

← Y_B

Secret key. → $K_a = (Y_B)^{x_A} \bmod q$

$K_b = Y_A^{Y_B} \bmod q$

Ex. The global element $q = 23$, & $\alpha = 9$. The node A has selected a private key 4 and B has selected a private key 3. Calculate public keys of A & B And generate secret key using Diffie Hellman key exchange algo.

$\alpha = 9, \quad q = 23$

$x_A = 4, \quad x_B = 3$

$Y_A = 9^4 \bmod 23$

$= 6$

$Y_B = 9^3 \bmod 23$

$= 16$

$K_a = (16)^4 \bmod 23$

$= 9$

$K_b = 6^3 \bmod 23$

$= 9$

DATE
PAGE NO.

Q. Consider a diffie Hellmans key exchange scheme with common prime $q = 11$ & primitive root $\alpha = 2$

1) Show that 2 is primitive root of 11

2) A has public key $Y_A = 9$. What is A's private key. public

3) If B has private key $X_B = 3$. what is the shared secret key K.

ii) $Y_A = \alpha^{X_A} \bmod q$

$9 = 2^{X_A} \bmod 11 \Rightarrow X_A = 6$

i)

$2^0 \bmod 11 = 1$
$2^1 \bmod 11 = 2$
$2^2 \bmod 11 = 4$
$2^3 \bmod 11 = 8$
$2^4 \bmod 11 = 5$
$2^5 \bmod 11 = 10$
$2^6 \bmod 11 = 9$
$2^7 \bmod 11 = 7$
$2^8 \bmod 11 = 3$
$2^9 \bmod 11 = 6$
$2^{10} \bmod 11 = 1$

iii) $3 = 2^{Y_A} \bmod 11 \Rightarrow Y_B = 8$

$K_B = (Y_A)^{X_B} \bmod q$

$= (9)^3 \bmod 11 = 3$

⇒ $q = 353$

$\alpha = 3$

$X_A = 97$, $X_B = 233$

Calculate Y_A, Y_B & K

$Y_A = 3^{97} \mod 353 = 40$

bi	1	1	0	0	0	0	1
ci	1	3	6	12	24	48	97
di	3	27	29	176	265	331	40

$Y_B = 3^{233} \mod 353 = 248$

bi	1	1	1	0	1	0	0	1
ci	1	3	7	14	29	58	116	233
di	3	27	69	172	149	315	32	248

$K = (Y_B)^{X_A} \mod q$

$= (40)^{233} \mod (248)^{97} \mod 353$

$= 160$

bi	1	1	0	0	0	0	1
ci	1	3	6	12	24	48	97
di	248	215	335	324	135	222	160

11/8

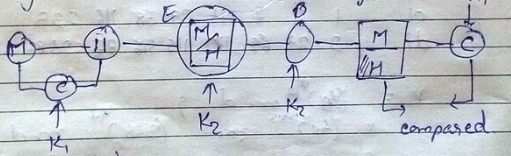
★ Message Authentication & Hash Function.

⇒ 1. Authentication
2. Confidentiality of message

- Content modification
- Sequence modification
- Timing modification.
- S/D Repudiation.

- Encry / Decry
- Message Auth. Code (MAC)
- Hash Function.

Msg auth with confidentiality.



⇒ Property of Hash
1) Weak collision resistance

$x = H(x) \quad y = H(y)$

$H(x) \neq H(y)$.

2) strong collision resistance

even input to hash fn is similar, output will be different.

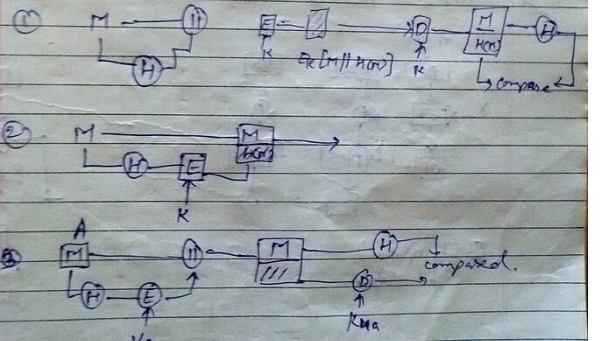
(x, y) $H(x) \neq H(y)$

$X = x_1 \oplus x_2 \oplus x_3 \oplus \ldots \oplus x_i$

$= (x_1 \oplus x_2 \oplus x_3 \oplus \ldots x_i) \,||\, H(x)$

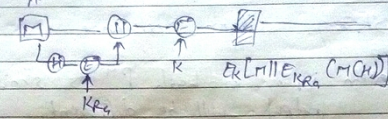
$Y = Y_1 \oplus Y_2 \oplus Y_3 \ldots \oplus Y_m$

Hash codes

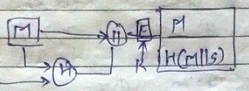


ensure authenticity alongs with any authentication ~~dont~~ dont ensure confidentiality.

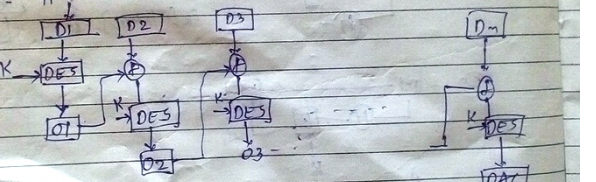
(5)



authentication, confidentiality & data integrity.



★ MAC Based DES



64 bit input
64-bit output.

Data Authentication Code

16 - 64 bit chosen

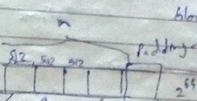
2/6/18

MD-5

(128 bits)

1) Append padding bits

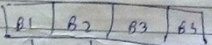
block size 512 bit

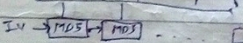


2) Append length bits

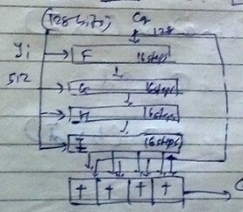
3) Initialize MD5 buffers

32 bit A : 01234567
B : 89abcdef
C : fedcba98
D : 76543210





General Block Diagram



+ addition mod 2^{32}

In single round only 1 word get modified

4. Process each 512-bit block.
5. Output message digest

DATE
PAGE No.

T = [ ]
64 value

$2^{32} \times abs(\sin(i))$
↑ radians

Compression function

$a \leftarrow b + ((a + g(b,c,d) + x[k] + T_i) <<< s)$

F, G, H, I

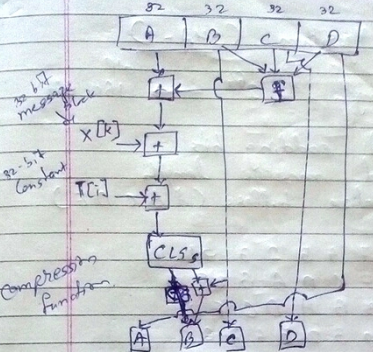
<<< s
↑ circular shift by s bits

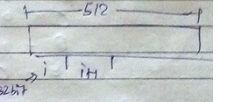
F(b,c,d) $(b \wedge c) \vee (\bar{b} \wedge d)$

G(b,c,d) $(b \wedge d) \vee (c \wedge \bar{d})$

H(b,c,d) $b \oplus c \oplus d$

I(b,c,d) $c \oplus (b \vee \bar{d})$





$X[i] =$

$X_{p_2}[i] = (1+5i) \bmod 16$ → 16 words

$X_{p_3}[i] = (5+3i) \bmod 16$

$X_{p_4}[i] = 7i \bmod 16$

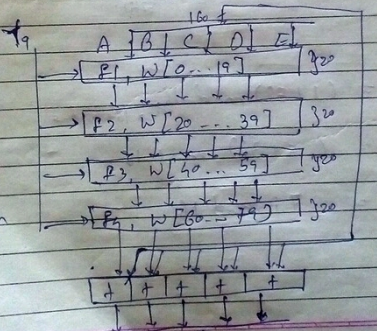
Q. 18

Secure Hash Algorithm (SHA-1)

Max. length $< 2^{64}$ ⟶ o/p 160 bit

(i) Append padding bits
(ii) append message length
(iii) Initialize buffers (A B C D E)

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



Generalized Block Diagram

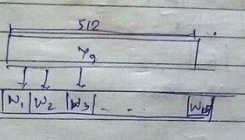
step no. | take int part of

$0 \le t \le 19$ $\quad K_t = 2^{30} \times \sqrt{2}$

$20 \le t \le 39$ $\quad K_t = 2^{30} \times \sqrt{3}$

$40 \le t \le 59$ $\quad K_t = 2^{30} \times \sqrt{5}$

$60 \le t \le 79$ $\quad K_t = 2^{30} \times \sqrt{10}$



For derive $W_{37} = W_1, W_3, W_9, W_{15}$

$W_{79} = W_{t-10}, W_{t-4}, W_{t-9}, W_{t-3}$

W_{t-3}

$= W_{63}, W_{65}, W_{71}, W_{76}$

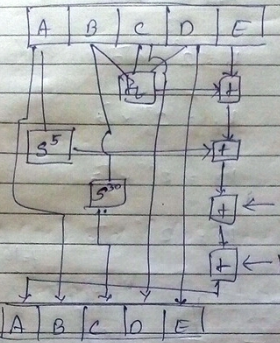
$F_1 = (B \wedge C) \vee (\bar{B} \wedge D)$

$F_3 = (B \wedge C) \vee (B \wedge D) \vee (C \wedge D)$

$F_2 = B \oplus C \oplus D$

$F_4 = B \oplus C \oplus D$

$f_t = f_1, f_2, f_3, f_4$



compression function

Fix value pre found in debird

single step of SHA1 Algorithm.

→ After 80 step single block is processed & o/p is message digest.

Properties

→ One way ness.

→ strong collision resistance &
+ different message → different hash
same message → different hash value

→ To ensure data integrity, we use message digest algo.

→ SHA-1 is more secure than MD5

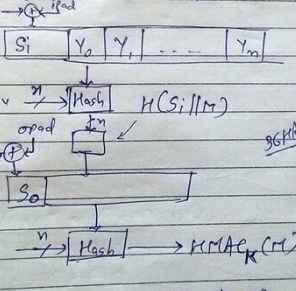
→ For crypto currency we use SHA-256

→ SHA-1 have more steps than MD5 but MD5 is faster

→ Little endian vs Big endian
(MD5) (SHA-1)

★ HMAC

Hashed Based MAC. → block size/length must be fixed.



K+ → key derived from original key

SHAK
ipad = 00110110 (b/e)
opad = 01011100 (b/e)
Repeat

→ K is secret key (Any length).
↳ 200, 300, 800 bits

if key > block length

K → [Hash] → Fix length → 160 + Padding (process)

if key < block length

key + padding → Process

$HMAC_K(M) = H\{(K^+ \oplus opad) \| H[(K^+ \oplus ipad) \| M]\}$

Syllabus HASH Algo MD5, SHA1 RSA,

(key exchange) key distribution public, secret