CS 434 → Cryptography and Security

03/08/2022

Marking:

- Mid Sem → 30
- End Sem → 50
- Sessional → 20
  - Minor Test → 5
  - Attendance → 5
  - Assignment → 10

~~Syllabus:~~

Cryptography + Crypt Analysis → Cryptology.

forming cryptic msg. / encoding → Encryption

breaking/decoding → Decryption without knowledge of the key. / Hacking/cracking.

- Encryption: Plain Text → Cipher Text
- ARX
- Caeser Cipher : $C = E(M) = (Msg + Key) \% 26$
  $M = D(C) = (C - Key + 26) \% 26$
  (Shift cipher)

Private key Encryption — Public key Encryption.

- Symmetric and Asymmetric
- Block Cipher and Stream Cipher.

→ Books: Chalice? Stalling? Forozone?

04/08/2022

* Security Attacks:

- Passive: No damage to data. Monitoring/Tapping data.
- Active: Modify/Damage original data.
  - Masquerade : Impersonating
  - Replay :
  - Modification :
  - Denial of Service : Irresponsive Server/Service by over-loading or flooding.

(Active Attacks)

__/__/____

* Security services:
  - → Authentication
  - → Confidentiality
  - → Integrity
  - → Non-repudation

* Symmetric key Encryption: 1 (private) key $PR_S = PR_R = PR$ Sender and Receiver.
  - Block cipher, Stream cipher
  - Same key for both (key distributio
* Asymmetric key Encryption: Each user has 2 keys.
  - Private key and Public key

PR_A	PU_A
PR_B	PU_B
⋮	⋮

Symmetric: $\underbrace{E_{PR}(M) = C}_{\text{Sender's side}}$; $\underbrace{M = D_{PR}(C)}_{\text{Receiver's side.}}$

Asymmetric:

Encryption side	Decryption side.
PR_A	PR_B
PU_A	PU_A
PU_B	PU_B

Sender

Authentication → (1) $E_{PR_A}(M) = C$; $M = D_{PU_A}(C)$

Confidentiality → (2) $E_{PU_B}(M) = C$; $M = D_{PR_B}(C)$

- Symmetric Encryption: Group of N users, ${}^nC_2$ keys required. $\frac{n(n-1)}{2}$.
- Asymmetric Encryption: Group of N users, 2n keys required.
- for both confidentiality and Authentication, combination of Asymmetric and Symmetric keys, C =

10/08/2022

- Monoalphabetic : 26!
- Playfair cipher : $[\,]_{5\times5}$

* Ciphertext only Attack
  Known plaintext
  Known ciphertext.
  Chosen ciphertext.

- Hill Cipher : $AA^{-1} = I$

$$C = PK \bmod 26$$

$$[C_1\ C_2 \dots C_n]_{1\times N} = [P_1\ P_2 \dots P_n]_{1\times N} \bullet \begin{bmatrix} K_{11} & K_{12} & \dots & K_{1n} \\ K_{21} & K_{22} & \dots & K_{2n} \end{bmatrix}_{N\times N}$$

Example : P_1 : hillcipher C_1 : hcrzssxnsp

$$[C_1\ C_2] = [P_1\ P_2]\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$$

Assuming, A→0 ; B→1 ; .... Z→25

for, hi ↔ hc ; $[7\ \ 2] = [7\ \ 8]K \bmod 26$

for ll ↔ rz ; $[11\ \ 11] = [17\ \ 25]K \bmod 26$

$$\begin{bmatrix} 7 & 2 \\ 11 & 11 \end{bmatrix} = \begin{bmatrix} 7 & 8 \\ 17 & 25 \end{bmatrix} K \bmod 26$$

$$\begin{bmatrix} 7 & 8 \\ 17 & 25 \end{bmatrix}^{-1} = \begin{bmatrix} 25 & 22 \\ 1 & 23 \end{bmatrix}$$

Symmetric Key Encryption.

→ Block Cipher : DES, AES
  → Feistel Network
  → S-P Network (substitution-permutation)
→ Stream Cipher

11/08/2022

- Vigenere Cipher: $P_1 \rightarrow C_1, C_2, \ldots C_n$ (with $K_1, K_2 \ldots K_n$)
- Vernam Cipher: $C_i = (P_i \oplus K_i)$
- One-time pad

- Transposition Techniques: HELLO → E L H O L
- Rail fence:
- NIT SILCHAR → N T I C A I S I H R

Symmetric Key Encryption Algorithm:

- DES
- AES
- S-DES

- Block size, key size, Number of Rounds, Sub key Generation, Round function.
- Confusion (cipher text vs key) and Diffusion (plaintext vs ciphertext)

S-DES: (Simplified DES)

Block Size: 8 bits
key Size: 10 bits
No. of Rounds: 2

Encryption

8 bit plaintext
↓
IP — initial permutation
↓
Round 1: f ← K_1
↓
S.W. — swapping
↓
Round 2: f ← K_2
↓
IP^{-1}
↓
8 bit Ciphertext

Decryption

8 bit Ciphertext
↓
IP
↓
f ← K_2
↓
S.W.
↓
f ← K_1
↓
IP^{-1}
↓
8 bit plaintext

__/__/____

Key generation

10 bit keys

P_{10} permutation: K_1 K_2 K_3 K_4 K_5 K_6 K_7 K_8 K_9 K_{10}

↑ Shuffle order:

10

5 5

LS-1 LS-1 left circulation

5 5

10

P_8 → K_1

5

LS-2 LS-2

5 5

} 5, 5 → 10. 2 bit is taken out making it 8 bit.

10

P_8 → K_2

P_8

Initial permutation: P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8

↑ Shuffle order.

Example: P_5 P_3 P_6 P_2 P_1 P_7 P_8 P_4

1 2 3 4 5 6 7 8

8

EP Expansion Permutation — E-table → 5 4 2 8 1 3 6 7

↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑

P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8

8

f ← K_1

8

∴ IP: 5 4 2 8 1 3 6 7

4 4

S_0 S_1 Sbox Substitution box

2 2

4

P_4

⊕

Expansion Permutation Example:

4 bit

b_4 | b_1 b_2 | b_3

b_2 | b_3 b_4 | b_1

L_1 R_1 = b_4 b_1 b_2 b_3 b_2 b_3 b_4 b_1

8 bit

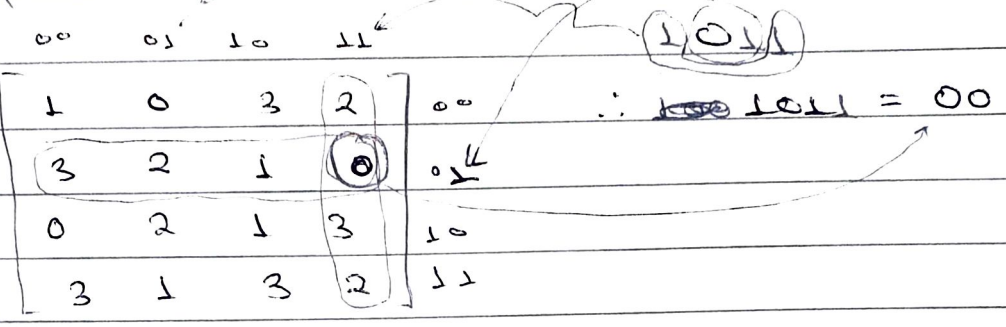
P_4

⊕

L_2 ← R_2

__/__/____

So Example:



* DES (Data Encryption Standard)

Block Size: 64 bit. Key Size: 64/56 bit No. of Rounds: 16

64 bit plaintext → 64

64 bit key.

$L_i = R_{i-1}$

$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$

Non-linear

Avalanche Effect.

In DES, even if just 1 bit is changed in plaintext, the output (cipher text) obtained will differ by 34 bit positions.

If 1 bit is changed in key, 35 bits in cipher text change.

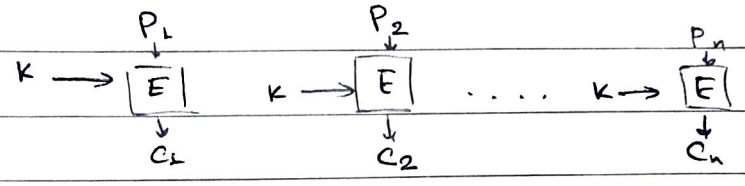
__/__/____

- Cipher Block Modes: (5 modes)

→ Electronic Code Block (ECB) Mode.

P_i : Plaintext of R

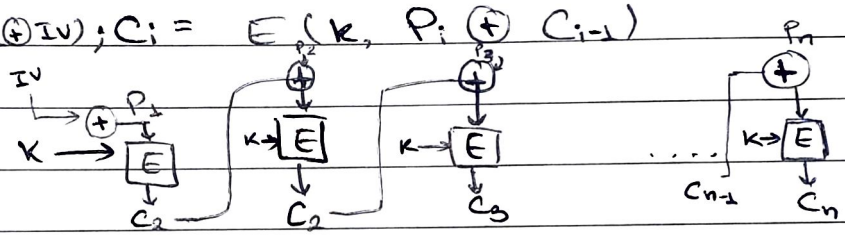
Encryption → $C_i = E(K, P_i)$



Decryption → $P_i = D(K, C_i)$

→ Cipher Block Chain (CBC) Mode

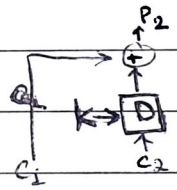
$C_1 = E(K, P_1 \oplus IV)$; $C_i = E(K, P_i \oplus C_{i-1})$



$P_i = C_{i-1} \oplus D(K, C_i)$

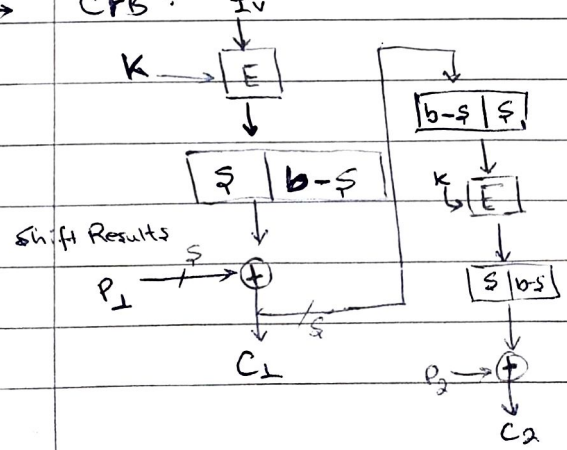
$x = P_2 \oplus C_1$

$P_2 = C_1 \oplus D(K, C_2)$



- → Cipher Feedback, Blockchain Mode (CFB):
- → Output Feedback Blockchain Mode (OFB):
- → Counter Mode

→ CFB:



$C_i = P_i \oplus MSB_s(O_i)$

where, $O_i = E(K, I_i)$

$I_i = LSB_{b-s}(I_{i-1})$

$P_i = C_i \oplus MSB_s(O_i)$

__/__/____

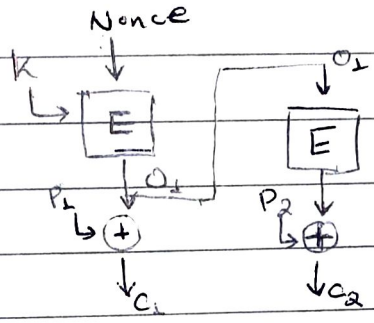
→ OFB:

Nonce.

$C_i = P_i \oplus O_i$

$O_i = E(K, I_i)$

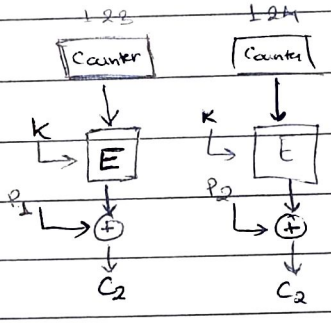
$I_i = O_{i-1}$



→ Counter Mode:

$C_i = P_i \oplus E(K, T_i)$

Counter



17/08/2022

* Differential Cryptanalysis:

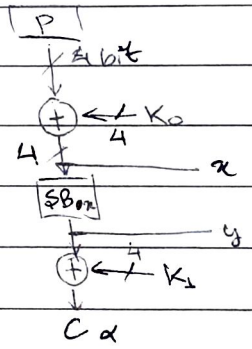
XOR diff.

Example: K = 8 → K_1 K_0

P_1 P_2

↓ ↓

C_1 C_2



S-Box.

0 → 3	6 → 5	11 → 2
1 → 14	7 → 6	12 → 13
3 → 10	8 → 8	13 → 12
4 → 4	9 → 11	14 → 0
5 → 9	10 → 15	15 → 7

Given, K = 1001 0110

9 6

K_1 K_0

If, $P_1 \to 10$ and $P_2 \to 12$.

find, C_1 and C_2.

for $P_1 \to 10_{10}$; $P_1 \to (1010)_2 \to$ 1010 ⊕ 0110 = $(1100)_2 = (12)_{10}$

$(13)_{10} = (1101)_2$ ⊕ 1001 = $(0100)_2 = (4)_{10}$

$\therefore P_1 \to 10 = C_1 \to 4$

__/__/____

for, $P_2 \rightarrow (12)_{10} \rightarrow (1100)_2 \rightarrow$ 1100

$\oplus$ 0110

$(1010)_2 \rightarrow (10)_{10} \rightarrow$ Step $\rightarrow (15)_{10}$

$(15)_{10} \rightarrow (1111)_2 \rightarrow$ 1111

$\oplus$ 1001

$(0110)_2 \rightarrow (6)_{10}$

$\therefore P_2 \rightarrow 12 = C_2 \rightarrow 6$

$\therefore (P_1, P_2) = (10, 12)$; $(C_1, C_2) = (4, 6)$

$P_1 \oplus K_0 = X$ $\qquad$ $P_2 \oplus K_0 = X'$

$X \oplus X' = P_1 \oplus P_2$

i/p $\alpha = x \oplus x'$

o/p $\beta = y \oplus y'$ $\qquad$ $\alpha = 6$; $\beta = ?$

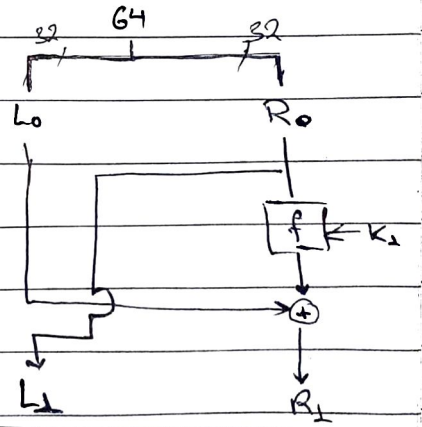
$y \oplus K_1 = C_1$ $\qquad$ $y' \oplus K_1 = C_2$

$y \oplus y' = C_1 \oplus C_2$

$\alpha = 6$; $\beta = 2$

19/08/2022

* Differential Cryptanalysis on DES:



$L_1 = R_0$

$R_1 = L_0 \oplus f(K_1, R_0)$

$L_i = R_{i-1}$

$R_i = L_{i-1} \oplus F(K_i, R_{i-1})$

$L_2 = R_1$

$R_2 = L_1 \oplus F(K_2, R_1)$

$L_3 = R_2$

$R_3 = L_2 \oplus F(K_3, R_2)$

$= L_0 \oplus F(R_0, K_1) \oplus F(R_2, K_3)$

$L_1^* = R_0^*$

$R_1^* = L_0^* \oplus F(K_1, R_0^*)$

$L_2^* = R_1^*$

$R_2^* = L_1^* \oplus F(K_2, R_1^*)$

$L_3^* = R_2^*$

$R_3^* = L_2^* \oplus F(K_3, R_2^*)$

__/__/____

$R_3 = L_0 \oplus F(R_0, K_1) \oplus F(R_2, K_3)$

$R_3^* = L_0^* \oplus f(R_0^*, K_1) \oplus f(R_2^*, K_3)$

$R_3' = R_3 \oplus R_3^*$

$R_3' = L_0' \oplus F(R_0, K_1) \oplus F(R_2, K_3) \oplus f(R_0^*, K_1) \oplus F(R_2^*, K_3)$

where, $L_0' = L_0 \oplus L_0^*$

Choosing plaintext pair such that $R_0 = R_0^*$

Then, $R_3' = L_0' \oplus F(R_2, K_3) \oplus f(R_2^*, K_3)$

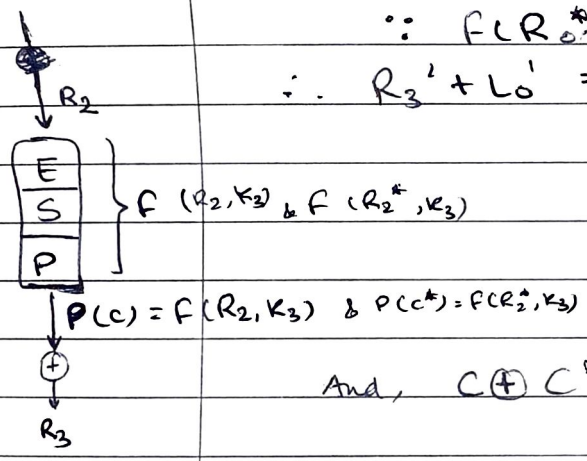
$\because f(R_0^*, K_1) \oplus F(R_0, K_1) = 0$

$\therefore R_3' + L_0' = f(R_2, K_3) \oplus f(R_2^*, K_3)$

$= P(C) \oplus P(C^*)$

permutation of C, permutation of C^*

C and C^* are outputs of S-Box.



And, $C \oplus C^* = C' = P^{-1}(R_3' \oplus L_0')$

Let, E and E^* are outputs of Expansion Permutation.

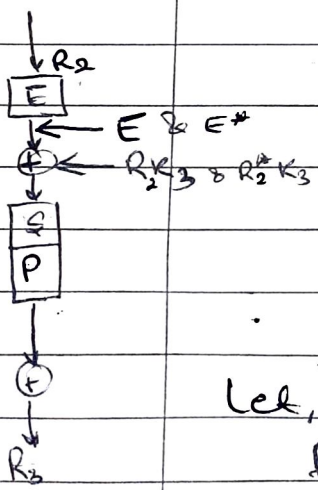
$L_3 = R_2$

$EP(R_2) = EP(L_3) = E$

$EP(R_2^*) = EP(L_3^*) = E^*$

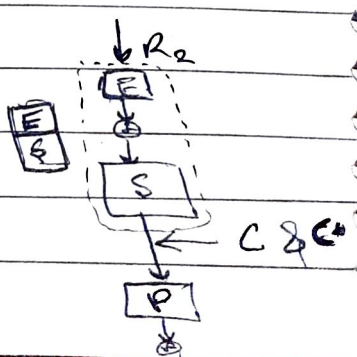
$B = E \oplus R_1 K_3 \; ; \; B^* = E^* \oplus R_2^* K_3$

$B \oplus B^* = E \oplus E^*$



Let, C and C^* are outputs of S-box

$f(R_2, K_3) =$



01/09/2022

LIETERSR

* Advanced Encryption Standard (AES) Conceptual Scheme:

- Block size: 128 bits.
  Key Size: 128 | 192 | 256
  No. of Rounds: 10 12 14

- Operations:
  - Add Round keys (xor ⊕)
  - Sub Bytes (S-box)
  - Shift Rows } Diffusion
  - mix Columns }

* $x^8 + x^4 + x^3 + x + 1$ $\quad b_i' = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i$

$c = 0\,1\,1\,0\,0\,0\,1\,(1)$ (c_0)

$b_0 = b_0 \oplus b_4 \oplus b_5 \oplus b_6 + b_7 \oplus 1$

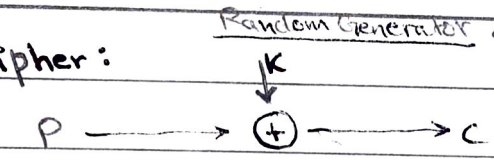
* Algebraic Cryptanalysis.

- Differential Cryptanalysis to break DES Keys.
- Algebraic Cryptanalysis to break AES keys.
  - → SAT model on Boolean.
  - → SMT model otherwise. : Z_3 Solver

Write rules / constraints of SMT.

02/09/2022

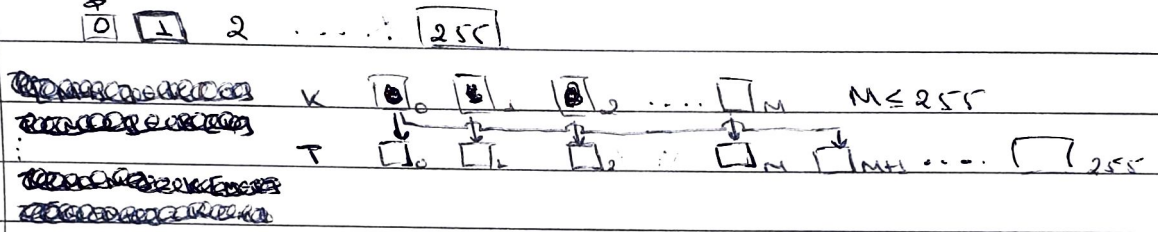
* **Stream Cipher:**



- High Periodicity : $C_i = P_i \oplus K_i$

* **RC4 Algorithm:**

Key → 0 - 255 bytes




```
for i = 0 to 255 do
    S[i] = i
    T[i] = K[i mod Keylength]
j = 0;
for i = 0 to 255 do
    j = (j + S[i] + T[i]) mod 256
    swap(S[i], S[j])

i, j = 0
while (true)
    i = (i+1) mod 256
    j = (j + S[i]) mod 256
    swap(S[i], S[j])
    t = (S[i] + S[j]) mod 256
    K = S[t]
```

* **LFSR: Linear Feedback Shift Resistor :** (To Generate bits randomly)

- Primitive Polynomial (irreducible)

__/__/____

Primitive polynomial →

3 bits: $f(x) = x^3 + x^2 + 1$ → $2^n - 1$

Here, + is ⊕ XOR

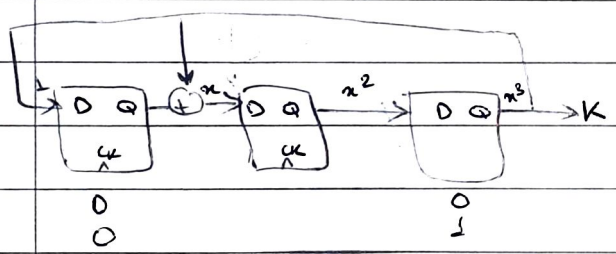
$x \equiv x$	010
$x^2 \equiv x^2$	100
$x^3 \equiv x^2 + 1$	101
$x^4 \equiv x^2 + x + 1$	111
$x^5 \equiv x + 1$	011
$x^6 \equiv x^2 + x$	110
$x^7 \equiv$	

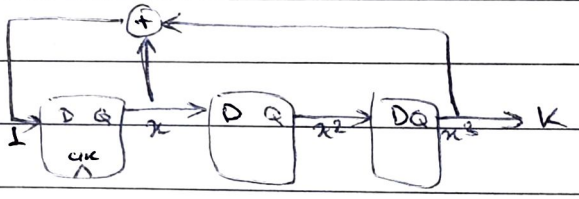
$x^3+x^2+1 \overline{)\, x^6}$ → x^3+x^2+x → x^2+x+1

Non-primitive $x^3 + x^2 + x + 1 < 2^n - 1$

irreducible: $x^4 + x^3 + x^2 + x + 1$

Feedbacks: Internal and External





1 1 1

⊕

1 0 1

1 0 0

0 1 0

P_2

P_1

P_0

... K_2 K_1 K_0 → ⊕

$P_0 \oplus K_0 = C_0$

$P_1 \oplus K_1 = C_1$

$P_2 \oplus K_2 = C_2$ ...

Take, $(C_0 - C_7)$ ... $(C_8 - C_{15})$ groups to perform xor operation

08/09/2022

* Extended Euclidean Algorithm:

$$d = \gcd(a,b) = ax + by$$

r is remainder
q is quotient.

e.g.: $\gcd(42,30) = 6$; So $42x + 30y = 6$; $x = -2$; $y = 3$

(a,b) $\quad a = q_1 b + r_1$ $\quad\quad r_1 = ax_1 + by_1$

(b,r_1) $\quad b = q_2 r_1 + r_2$ $\quad\quad r_2 = ax_2 + by_2$

(r_1,r_2) $\quad r_1 = q_3 r_2 + r_3$

$\vdots$

(r_{n-1}, r_n) $\quad r_{n-1} = q_{n+1} r_n + 0$ $\quad\quad r_n = ax_n + by_n$

So, $r_{i-2} = q_i r_{i-1} + r_i$

i.e., $r_i = r_{i-2} - q_i r_{i-1}$ $\quad\quad r_{i-2} = ax_{i-2} + by_{i-2}$

$$r_i = (ax_{i-2} + by_{i-2}) - q_i (ax_{i-1} + by_{i-1})$$
$$= a(x_{i-2} - q_i x_{i-1}) + b(y_{i-2} - q_i y_{i-1})$$
$$= ax_i + by_i$$

$r_{-1} = a$; $y_{-1} = 0$; $x_{-1} = 1$
$r_0 = b$; $y_0 = 1$; $x_0 = 0$

where, $x_i = x_{i-2} - q_i x_{i-1}$

$y_i = y_{i-2} - q_i y_{i-1}$

Example

gcd (1759, 550)

i	r_i	q_i	x_i	y_i
-1	a 1759		1	0
0	b 550		0	1
1	109	3	1	-3
2	5	5	-5	16
3	4	21	106	-339
4	1	1	-111	355
5	0	4		

So, $a = 1759$; $b = 550$; $ax + by = d = \gcd(a,b)$

or, $1759 \times (-111) + 550 \times 355 = 1$ $\quad\quad \therefore d = 1$

__/__/____

quotients.

- Euler Totient : $\phi(n)$: No of Relative Primes.

  If $\gcd(a,b) = 1$; a and b are relative primes.

  $\phi(p) = p-1$ primes ; $p \rightarrow$ prime

  E.g.: $\phi(7) = 1, 2, 3, 4, 5, 6,$ = 6 primes.

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

  where, $n = p*q$; $p, q \rightarrow$ primes.

  E.g.: $\phi(15) = (3-1) * (5-1) = 8$ primes.

  $\phi(15) = 1, 2, 4, 7, 8, 11$

- Fermat's Theorem:

  $a^{p-1} \equiv 1 \bmod p$; $p \rightarrow$ prime

  a is an integer not divisible by p

E.g.: $a = 7$; $p = 19$.

$a^{p-1} \equiv 7^{18} \bmod 19$

$\equiv 1$

$11 \equiv 7^2 \bmod 19$

$7 \equiv 7^4 \bmod 19$

14/09/2022

* Primality Test (→ Miller-Rabin)

If 'p' is a prime number, then $p-1 = 2^k \times q$ is even num.

i.e, if $p = 561$ then, $560 = 2^4 \times 35$

Step I: $p-1 = 2^k \times q$; $k = ?$, $q = ?$

Step II: Choose 'a' such that $1 < a < p-1$

Step III: Compute. $b_0 = a^q \bmod p$

$b_1 = a^{2q} \bmod p$

$b_i = b_{i-1}^2 \bmod p$

if b_i is +ve 1, $+1 \rightarrow$ Composite

-ve 1, $-1 \rightarrow$ Probable Prime.

Example; $p = 561$; $p-1 = 560 = 2^4 \times 35$; $k = 4$, $q = 35$; Let, $a = 2$

Then, $2^{35} \bmod 561 \equiv 263 = b_0$

$263^2 \bmod 561 \equiv \quad = b_1$

__/__/____

Recursive

$b_0 = 2^{35} \bmod 561 \equiv 263$

$b_1 = 263^2 \bmod 561 \equiv 166$

$b_2 = 166^2 \bmod 561 \equiv 67$

$b_3 = 67^2 \bmod 561 \equiv 1$

So, 561 is a compositive number.

* Check if 2047 is prime or not.

Let: 2047 be prime, P. Then, $P-1 = 2046$. Let $a = 2$.

$2046 = 2^{1023} \bmod 2047$

So, $b_0 = 2^{1023} \bmod 2047$

$\equiv (2^{11})^{93} \bmod 2047$ $(\because 2^{11} \bmod 2047 \equiv 1)$

$\equiv 1^{93} \bmod 2047$

$\equiv 1$

$\therefore$ 2047 is a compositive number.

* $\phi(n)$ Euler Quotient. If P is Prime: $\phi(P) = P-1$ relative primes. ← Studied in previous class.
* Euler's Theorem: $a^{\phi(n)} \equiv 1 \bmod n$

where, a, n are relative primes.

Example: $a = 3$; $n = 10$. Then

$a^{\phi(10)} \equiv 1 \bmod n$

or, $3^4 \equiv 1 \bmod 10$

or, $81 \bmod 10 \equiv 1$ True.

* Prime factors:

Any number n canbe represented as the product of prime numbers i.e., $n = p_1^k \times p_2^k \times \ldots$

Eg: $n = 300 = 2^2 \times 5^2 \times 3$

2	300
2	150
5	75
5	15
3	3

15/09/2022

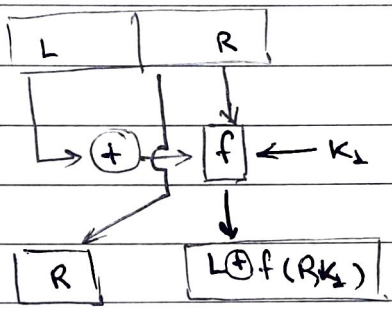
46) * Suppose the DES f function mapped every 32-bit input R, regardless of the value of the input K, to:

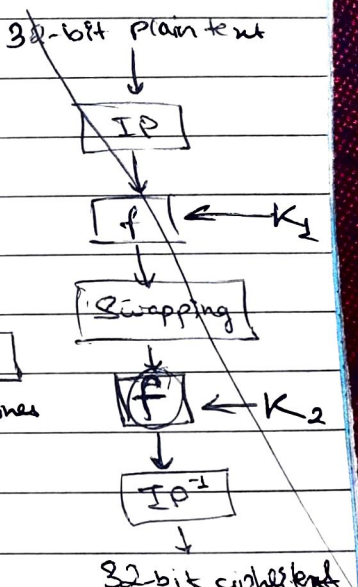
a. 32-bit string of ones. b. bitwise complement of R.

Then,

1. What function would DES then compute?
2. What would the decryption look like?

➤ 32-bit input ; ~~key~~ Round key size : 48 bit





$f(R,K_1)$ = 1 1 1 1 1 .... (32 bit ones)

$L \oplus f(R,K_1) = L'$

1st step : R L'

2nd step : L' $R \oplus f(L',K_1) = R'$

3rd step : R' $(L')' = L$

4th step : L $(R')' = R$

$IP^{-1}(RL)$

21/09/2022

Q. Using fermat's theorem, find $3^{201} \bmod 11$.

→

$3^{201} \bmod 11 \equiv 3$

from fermat's theorem, we know,

$3^{10} \bmod 11 \equiv 1$

So, $3^{201} \bmod 11 \equiv (3^{10})^{20} \times 3 \bmod 11$

$\equiv 1 \times 3 \bmod 11$

$\equiv 3$

$(3)^{10} \equiv 1$

$(3^{10})^{20} \equiv 3$?

$((3^{10})^{20} \times (3^1)) \bmod 11$

$\equiv 1 \times 3 \bmod 11$

$\equiv 3$

Q. $\phi(231) = (3-1) \times (7-1) \times (11-1) = 2 \times 6 \times 10 = 120$

$231 = 3 \times 7 \times 11$

Q. if, $3x \equiv 4 \bmod 5$

$9x \equiv 3 \bmod 7$

$7x \equiv 6 \bmod 9$; find x.

CRT

Soln:-

$x \equiv 3^{-1} 4 \bmod 5$

$\equiv 8 \bmod 5$

$\therefore x \equiv 3 \bmod 5$

$3^{-1} \bmod 5 \equiv 2$

So, $2 \times 4 \bmod 5 = 8 \bmod 5 = 3$

Similarly, $x \equiv 9^{-1} 3 \bmod 7$

$\equiv$

$\equiv 5 \bmod 7$

And, $x \equiv 7^{-1} 6 \bmod 9$

$\equiv$

$\equiv 6 \bmod 9$

So,

$x = 33$

E	N	C	R	Y	P	T	I	O	N
4	13	2	17	24	15	19	8	14	13

Vigenere Cipher * p: Encryption leg

key: ~~key~~ ~~key~~ ~~key~~ ~~key~~ ~~key~~ leg

$C_i = (P_i + K_i \bmod m) \bmod 26$

$C_i = (P_i + K_{i \bmod m}) \bmod 26.$

l e g M = 3

K_0 K_1 K_2

Soln:-

$C_0 = (P_0 + K_0 \bmod 3) \bmod 26$

$C_0 = (4+11) \bmod 26 = 15 = P$

$C_1 = (13+4) \bmod 26 = 17 = R$

$C_2 = (2+6) \bmod 26 = 8 = I$

$C_3 = (17+11) \bmod 26 = 2 = C$

$C_4 = (24+4) \bmod 26 = 2 = C$

$C_5 = (15+6) \bmod 26 = 21 = V$

$C_6 = (19+11) \bmod 26 = 5 = E$

$C_7 = (8+4) \bmod 26 = 12 = M$

$C_8 = (14+6) \bmod 26 = 20 = U$

$C_9 = (13+11) \bmod 26 = 24 = Y$

$\therefore C = PRICCVEMUY$

PRICCVEMUY

* Let x' be the bitwise complement of x. Prove that

$C = E(P, k)$

$C' = E(P', K')$

8-bit : 3 2 1 4 5 7 8 6

64-bit

P → IP

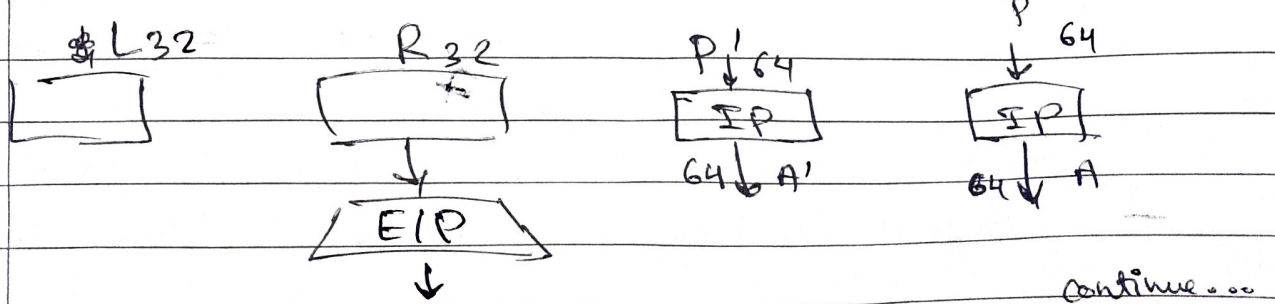
P 1011 0011

1 2 3 4 5 6 7 8

IP 1011 0110

C' → P 0100 1100

IP 0100 1001



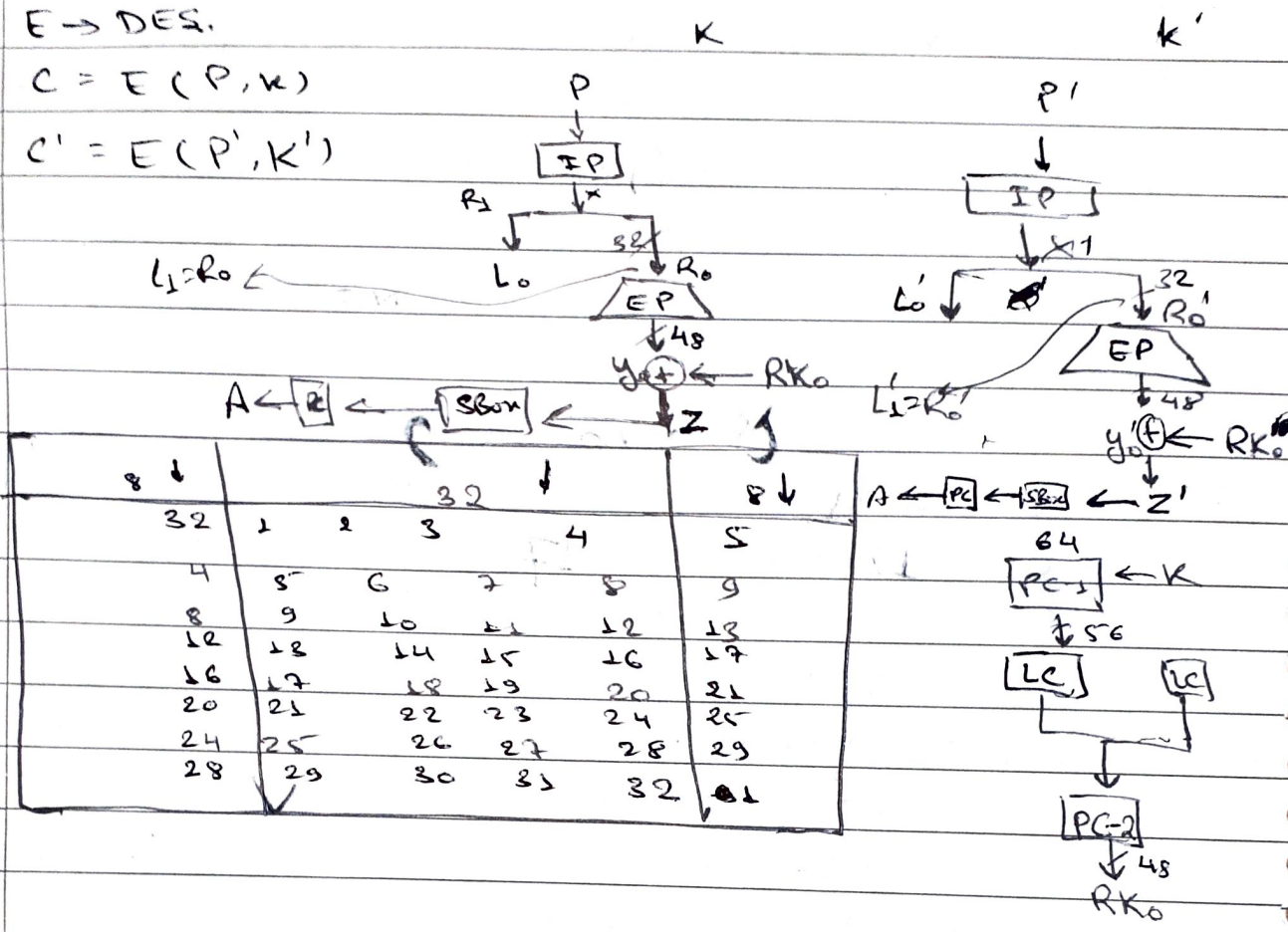
continue...

22/09/2022

* E → DES.

$C = E(P, K)$

$C' = E(P', K')$



8 ↓	32 ↓				8 ↓
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

* AES Algorithm

Plaintext, P = {00 01 02 03 04 05 .... 0E 0F} 32-Hexa 128-bin.

Key = {01 01 01 01 01 01 .....} 32 Hex 128bit bin

State 4×4. Show original content of the msg?

128 bits Key = 128 | 192 | 256

Rounds = 10 | 12 | 14

State = 4×4

00	04	08	0C
01	05	09	0D
02	06	0A	0E
03	07	0B	0F

4×4

In AES first Round is same as original plain.

Show the value of state after initial Round Key.

- First Round key is same as original key.

__/__/____

- After adding rounds: new state,

with

01	05	09	0D
00	04	08	0C
03	07	0B	0F
02	06	0A	0E

Add previous round with key.
i.e, Round ⊕ key.
0C⊕01 = 0D
01⊕01 = 00
02⊕01 = 03 and so on.

~~Adding Round key with~~ New S-Box.

7C	6B	01	D7
63	F2	30	FE
7B	C5	2B	76
77	6F	67	AB

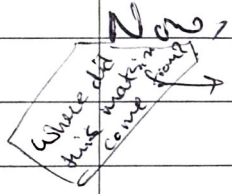
(AES S-box)

Left Shifting:

Shifted 0 times ←	7C	6B	01	D7
Shifted 1 times ←	F2	30	FE	63
Shifted 2 times ←	2B	76	7B	C5
Shifted 3 times ←	AB	77	6F	67

Shift x times
where, x is row index
x times to the left.

Now,



$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \times \begin{bmatrix} 7C \\ F2 \\ 2B \\ AB \end{bmatrix}$$

dot is polynomial multiplication

02 · 7C + 03 · F2 + 01 · 2B + 01 · AB

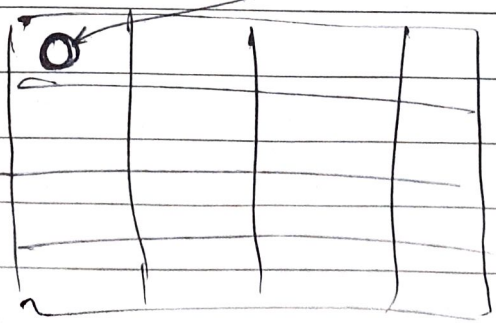
02 = 0000 0010 → x

7C = 0111 1100 → $x^6+x^5+x^4+x^3+x^2$

03 = 0000 0011 → $x^1+x^0 = x+1$

01 = 0000 0001

= ?



similarly, for all other boxes, we need to calculate respective polynomial multiplication

Unit 4

Asymmetric Key Cipher and Hashing

12/10/2022

* Public Key Cryptography: (ASymmetric Key)

A	B
PR_A	PR_B
PU_A	PU_B

Sender Receiver

i. $PR_A \rightarrow PU_A$ (must) → Authentication

ii. $PU_B \rightarrow PR_B$ (must) → Confidentiality

× $PR_A \not\rightarrow PR_B$

× $PU_A \not\rightarrow PU_B$

ie, if at sender's side, private key is used, it can be decrypted using only public key and vice versa, if public key is used during encryption, then it can be decrypted using private key only.

$E_{PR_A}[P] = C$
$D_{PU_A}[C] = P$ } Authentication.

$E_{PU_A}[P] = C$
$D_{PR_A}[C] = P$ } Confidentiality

$E_{PU_B}(E_{PR_A}(P)) = C$
$D_{PU_A}(E_{PR_B}(C)) = P$ } Both Authentication and Confidentiality

For N users,

Private key = $^{N}C_2$

Public key = $2N$

__/__/____

- RSA Algorithm :

Steps :

- Key generation :

SELECT — Select p, q , where p, q are primes, $p, q \neq 2$; $p, q > 2$ and, $p \neq q$

CALCULATE — Calculate, $n = p * q$

CALCULATE — Calculate, $\phi(n) = (p-1) * (q-1)$

SELECT — Select 'e' such that $\gcd(e, \phi(n)) = 1$

CALCULATE — Calculate 'd' ; such that, $e * d = 1 \bmod \phi(n)$

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

Sender, $PU = \{e, n\}$

Receiver, $PR = \{d, n\}$

ENCRYPTION

$C = P^e \bmod n$

DECRYPTION

$P = C^d \bmod n$

Example : Let, $p = 3$, $q = 11$, $m = 6$, $e = 7$ (given)

find $d = ?$, $c = ?$

Soln :

$n = p * q = 3 * 11 = 33$

$\phi(n) = 2 * 10 = 20$

$\gcd(e, 20) = 1$ when, $e = 7$ (given).

So, $d = 7^{-1} \bmod 20$

14/10/2022

$a^x = b$; $\$ x = \log_a(b)$

$y_A = \alpha^{X_A} \bmod q$ $\quad$ $y_B = \alpha^{X_B} \bmod q$

$K = \alpha^{X_A X_B}$

$C = M^e \bmod n$

$n = p * q$

discrete log ? dlog

$b = a^x \bmod p$; $x = \text{dlog}_{(a,p)}\, b$

which is huge number / iteratives / attempts?
Hence, cracking is impossible / difficult

Q. $8 = 5^x \bmod 13$, find x.

$x = \text{dlog}_{(5,13)}\, 8 = 3.$

$5^3 \bmod 13 ; \equiv 3$

$5^7 \bmod 13 ; \equiv 8$

$5^{11} \bmod 13 ; \equiv 8$

multiple solutions.
Hence, not reliable; not secure
not secure

Must have unique solution. : Take $p \rightarrow$ prime.
$\alpha \rightarrow$ primitive root.

$\phi(\phi(n))$ is the number of primitive roots.

$\phi(n) \rightarrow$ primitive primes; $\phi(\phi(n)) \rightarrow$ primitive roots.

- If, $n = p * q$; p and q are primes
such that $p \simeq q$ (nearly equal)
and $p < q$.

Then, Time Complexity to find p & q $= O(p)$
$= O(\sqrt{n})$

where, $n = p^2$

$\because$ when, $p \simeq q$, $p * q \simeq p^2$

__/__/____

✱ Elgamal Crypto System:

- public components:

  $q \rightarrow$ prime.

  $\alpha \rightarrow$ primitive root ; $\alpha < q$.

- Sender/Receiver A (lice)
  - $X_A \rightarrow$ Secret key
  - CALCULATE $Y_A = \alpha^{X_A} \bmod q$
  - Public key $\{q, \alpha, Y_A\}$
  - Private key $\{X_A\}$

- Receiver/Sender B (ob)

If B sends message to A,

Plaintext $\rightarrow M \quad < q$

Secret key $\rightarrow X_B \quad < q$

Calculate, $K = Y_A^{X_B} \bmod q$

$C_1 = \alpha^{X_B} \bmod q$

$C_2 = KM \bmod q$

(C_1, C_2)

When A decrypts B's message,

(α, q, Y_A, X_A)

(C_1, C_2)

$Y_A^{X_B} = (\alpha^{X_A})^{X_B} = (\alpha^{X_B})^{X_A} = (C_1)^{X_A}$

$K = C_1^{X_A} \bmod q$

$M = C_2 K^{-1} \bmod q.$

Questions: If,

$q \rightarrow 19$

$\alpha \rightarrow 10$

$X_A \rightarrow 5$

$M \rightarrow 17$

$Y_A \rightarrow ?$

$X_B \rightarrow 6$

$Y_B \rightarrow ?$

$C_1 \rightarrow ?$

$C_2 \rightarrow ?$

$Y_A = 10^5 \bmod 19 =$

$K = Y_A^6 \bmod 19 =$

$C_1 = 10^6 \bmod 19 =$

$C_2 = K * 17 \bmod 19 =$

26/10/2022

* ECC: Elliptic Curve Cryptography:

- Public key cryptography.
- Faster than RSA
  - 80 bits of key symmetric
    - 1024 bits → RSA
    - ~160 bits → ECC

- Equation: Set of points.

$E(a,b): y^2 = x^3 + ax + b$

condition: $4a^3 + 27b^2 \neq 0$

for finite, mod p: $y^2 = x^2 + ax + b \rangle \text{mod } p$

$E_{11}(1,1) = y^2 = x^3 + x + 1 \quad \text{mod } 11$

~~Example: point given (2,4) ; i.e. (x,y) = (2,4)~~

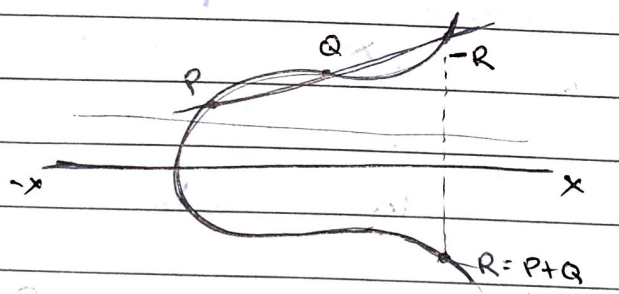
~~$4^2 =$~~

Eg.: $E(3,2): y^2 = x^3 + 3x + 2$

for point (2,4) ; i.e., $(x,y) = (2,4)$

So, $4^2 = 2^3 + 3 \times 2 + 2$

$\therefore 16 = 16$. Hence, it lies on the curve.



$P + Q = R$

$P \neq Q$

$P(x_1, y_1)$

$-P(x_1, -y_1)$

If, $P(x_1, y_1)$

$Q(x_2, y_2)$

$R(x_3, y_3) = P + Q$

$x_3 = \lambda^2 - x_1 - x_2 \quad \text{mod } p$

$y_3 = \lambda(x_1 - x_3) - y_1 \quad \text{mod } p$

$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \quad \text{mod } p$

(mod p : it is on finite field)

__/__/____

Example: $P(3,8)$, $Q(6,5)$, $E_{11}(1,1)$

find, $R = P + Q = ?$

Soln:- $\lambda = \frac{6-3}{5-8} \bmod 11 = \frac{3}{-3} \bmod 11 = -1 \bmod 11$

$\therefore \lambda = 10$

$x_3 = (10)^2 - 3 - 6 \bmod 11$

$= 100 - 9 \bmod 11$

$= 91 \bmod 11$

$\therefore x_3 = 3$

$y_3 = 10(3-6) - 8 \bmod 11$

$= -30 - 8 \bmod 11$

$= -38 \bmod 11$

$\therefore y_3 = 2$

$\therefore R(3,2)$ •

In, $E_{11}(1,1)$: • $y^2 = x^3 + x + 1 \bmod 11$

$R(3,2)$: $2^2 = 3^3 + 3 + 1 \bmod 11$

$4 = 27 + 4 \bmod 11$

or, $4 = 31 \bmod 11$

$\therefore 4 = 4$.

Hence, R(3,2) lies on the curve.

When $P = Q$,

$$\lambda = \frac{3x_1^2 + a}{2y_1} \bmod p.$$

When, $P \neq Q$,

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \bmod p$$

E-g:. If $P(4,6) \Rightarrow 2P = ?$ Given, $E_{11}(1,1)$.

$E_{11}(1,1)$: $y^2 = x^3 + x + 1 \bmod 11$

$P(4,6)$: $6^2 = 4^3 + 4 + 1 \bmod 11$

$36 \bmod 11 = 64 + 5 \bmod 11$

$3 = 69 \bmod 11$

$\therefore 3 = 3$

$2P = (6,6)$ Process?

verify: $(6^2 = 6^3 + 6 + 1) \bmod 11 \Rightarrow 3 = 3$. Yes, 2P (6,6) lies on the elliptic curve.

__/__/____

$Q = nP$. Given n, it is easy to find Q or P.
But, If only Q and P given, it is difficult to find n. Hence, n is our secret key.

- Find different points on the elliptic curve with eqn:

$E_p(a,b) = E_{11}(1,1) : y^2 = x^3 + x + 1 \bmod 11$

Since mod 11 is used, all point values must be under 11.

So,

$x=y=n$	$y^2 \bmod 11$	$x^3 + x + 1 \bmod 11$
0	0	1
1	1	3
2	4	0
3	9	9
4	5	3
5	3	10
6	3	3
7	5	10
8	9	4
9	4	2
10	1	10

Possible Inputs for x-value and y-value.

So, valid points: (0,1), (0,10), (4,5), (4,6), (2,0), (3,8), (8,2), (6,5), (6,6), (3,3).

- Using these principles, how to encrypt a message?

Public: $E_p(a,b)$

G : →

	A	B
Private:	n_A	n_B
Public:	$PU_A = n_A G$	$PU_B = n_B G$
Secret key,	$K = n_A PU_B$	$K = n_B PU_A$

__/__/____

Sender: $C = (C_1, C_2)$; $C_1 = n_A G$ (Public)

$C_2 = P + n_A P_{U_B}$

Receiver: $P = C_2 - n_B C_1$; $C_2 = P + n_A P_{U_B}$

$C_1 = n_A G$

Example: $E_{11}(1, 1)$; Public Point (Generator), $G(1, 5)$

Assume, $n_A = 2$,

Then, find $2(1, 5)$; i.e, $2G$.

02/11/2022

In the paper attached.

09/11/2022

★ Hash Function Uses:

- Message Integrity Check (MIC)
- .
- .
- .

★ Hash functions and Message Authentication.

- ~~[illegible]~~

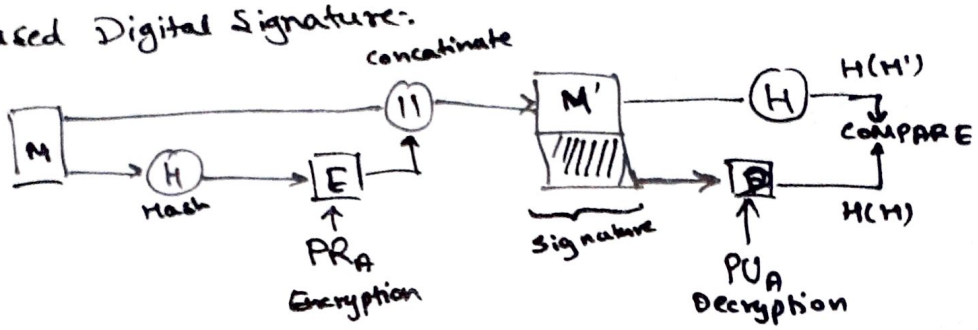
- Other Hash function Uses:
  - Pseudorandom function (PRF)
    - ↳ Generate session keys, nonces
    - ↳ Produce key from password.
    - ↳ Derive keys from master key cooperatively
  - Pseudorandom number generator (PRNG)
    - ↳ Vernam cipher / OTP,
      Proof of 'what you have' via SMS/Message
  - to create one-way password file
  - for intrusion detection and virus detection.

02/11/2022 : Crypto

* Digital Signature:

- RSA based DS.
- DSA based D.S.
- ECDSA based D.S..

- RSA Based Digital Signature:



- DSA Based Digital Signature

Public Components

p - prime ; $2^{L-1} < p < 2^L$; 512 bit ≤ L ≤ 1024

q - prime divisor of (p-1)

g - $h^{(p-1)/q} \bmod p > 1$; 'h' is any integer.

Sender

PR_A - Random Num x ; $0 \le x \le q$ - secret

PU_A - $y = g^x \bmod p$ - public

M - msg
H(m) - Hash of msg
M' - modified msg

Signing process

Select random int 'k' for each msg.

$r = (g^k \bmod p) \bmod q$

$s = k^{-1} (H(m) + x \cdot r) \bmod q$

(r, s) signature.

Receiver / Verification Process:

$w = s^{-1} \bmod q$

$u_1 = H(M') \cdot w \bmod q$

$u_2 = r \cdot w \bmod q$

$v = (g^{u_1} y^{u_2} \bmod p) \bmod q$

$r'V = r'$ sign void ?

$$g^{u_1} \times g^{u_2} = g^{u_1} \times g^{u_2 x} = g^{u_1 + u_2 x}$$
$$= g^{H(m)\,s^{-1} + r s^{-1} x}$$
$$= g^{s^{-1}(H(m) + xr)}$$

* ECDSA:

PUBLIC

q - prime number.

a, b : $y^2 = x^3 + ax + b$

G (x_g, y_g)

n - order.

Sender

PR : d random integers, $d < n-1$

PU : $Q = dG$

Signing

random 'k' ; $k < n-1$ for each msg.

$P = (x, y) = kG$

$r = x \bmod n$

$s = k^{-1}(H(m) + dr) \bmod n$; $s \neq 0$

Verification (r, s)

$w = s^{-1} \bmod n$

$u_1 = H(m)\,w \bmod n$

$u_2 = r\,w \bmod n$

COMPUTE : $(x_1, y_1) = u_1 G + u_2 Q$

$x_1 \bmod n = r$? sign void

$$u_1 G + u_2 Q = u_1 G + u_2 dG = H(m)\,s^{-1} G + r s^{-1} dG$$
$$= s^{-1}(H(m) + rd)\,G$$
$$= kG$$

__/__/____

Sender: $C = (C_1, C_2)$; $C_1 = n_A G$ (Public)
$C_2 = P + n_A P_{U_B}$

Receiver: $P = C_2 - n_B C_1$; $C_2 = P + n_A P_{U_B}$
$C_1 = n_A G$

Example: $E_{11}(1, 1)$; Public Point (Generator), $G(1, 5)$

Assume, $n_A = 2$,
Then, find $2(1, 5)$; i.e, $2G$.

02/11/2022

In the paper attached.

09/11/2022

★ Hash Function Uses:
- Message Integrity Check (MIC)
.
.
.

★ Hash functions and Message Authentication.
- ~~[illegible]~~

- Other Hash Function Uses:
  - Pseudorandom function (PRF)
    - Generate session keys, nonces
    - Produce key from password.
    - Derive keys from master key cooperatively
  - Pseudorandom number generator (PRNG)
    - vernam cipher/OTP,
      proof of 'what you have' via SMS/message
  - to create one-way password file
  - for intrusion detection and virus detection.

__/__/____

- Two Simple Insecure Hash Functions:

- Hash function Requirements.
  - variable input size
  - Fixed output size
  - Efficiency
  - Preimage resistant
  - Second preimage resistand.
  - Collison resistant
  - Pseudorandomness.

- Attacks on Hash functions

- Birthday Attacks: Birthday Paradox.
- SHA versions:

	SHA-1	-224	-256	-384	-512
Msg digest size	160	224	256	384	512
Block Size	512	512	512	1024	1024
Msg Size					
word Size					
No of Rounds Steps.					

Key Management Unit 5

15/11/2022.

Cryptography and Network Security - Ch-14.
Fifth edition by William Stallings.

- Road Map -
  - Symmetric key distribution using Symmetric encryption
  - Symmetric key distribution using public-key encryption
  - distribution of public keys

Symmetric key distribution using Symmetric Encryption: ↓

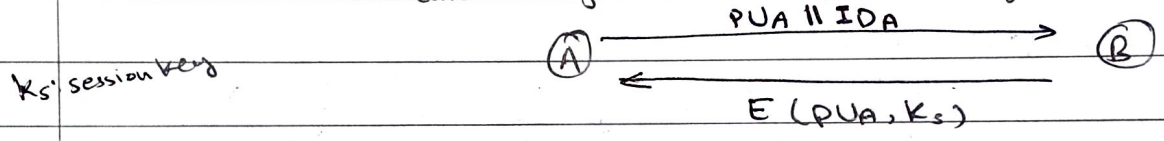
- Key distribution
  - A can select key and physically deliver to B.
  - Third party can select and deliver key to A & B.
  - If A and B have communicated before, they can use the same keys as before (or previously used).
  - If A and B have secure communication with third party C, C can relay messages between A and B.

- **Key Hierarchy** → Session key → Temporary, used for encryption of data between two users, for one logical session
  → Master key

- **Key distribution Issues:**
  - trust on KDCs (key distribution centres).
  - Session key lifetimes should be limited.
  - Automatic key distribution on behalf of users should be trusted system
  - controlling key usage.
  - Decentralised key distribution.

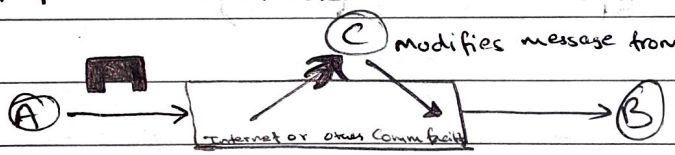
* **Symmetric key Distribution Using Public keys: ↓**
- Public key cryptosystems are inefficient.

- **Simple Secret key distribution:**
  - Merkel proposed very simple scheme:
    - allows secure comm.
    - no keys before/after exist.

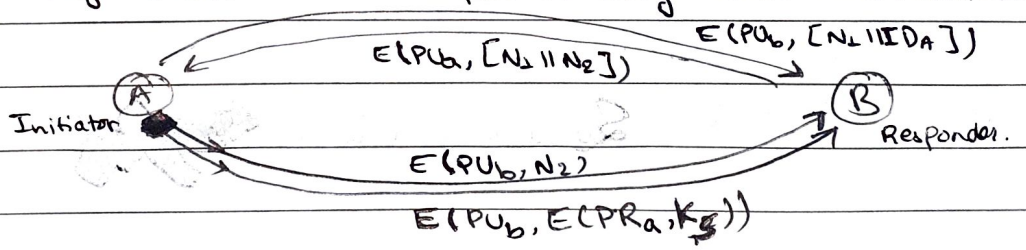


K_s: session key

- **Man in the Middle Attack:**



- **Secret key Dist. with Confidentiality and Authentication.**



- **Distribution of Public Keys:**
  - PUBLIC ANNOUNCEMENT
  - Publicly Available Directory.
  - PUBLIC KEY AUTHORITY.
  - PUBLIC KEY CERTIFICATES

16/11/.

Ch-15 : User Authentication:

- Kerberos:
  - Authentication service developed as a part of Project Athena at MIT.
  - 'Multi-headed dog'.
  - centralised authentication server

Kerberos V.4:
- makes use of DES to provide user authentication service.
  - → Authentication service exchange to obtain ticket-granting ticket
  - → Ticket-granting service Exchange to obtain service-granting ticket
  - → Client/Server Authentication Exchange to obtain service.

Unit 6 Security

Email Security

- Email Security Enhancements:
  - Confidentiality: protection from disclosure
  - Authenticity: of sender of message
  - Message integrity: protection from modification.
  - Non-repudiation of origin: protection from denial by sender.

- Pretty Good Privacy (PGP)
  - PGP Operation → Authentication (SHA)
  - PGP Operation → Confidentiality (CAST)
  - PGP op → Confidentiality and Authentication
  - PGP op → compression
  - PGP op → Email compatibility (RADIX-64 Algo)
  - PGP op → Summary

17/11/2022

SSL: Secure Socket Layer

To be discussed later!

17/11/2022

* C=10; n=35; e=5; P=? (RSA Algorithm)

→ $P = C^d \bmod n$

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

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

$n = p*q$, where p and q are primes

$\phi(n) = 24$; $ed = 1 \bmod \phi(n)$

$5 \times d = 1 \bmod 24$

$\therefore d = 29$

$P = 10^{29} \bmod 35 =$

Sender	Receiver
e, n	d, n

Q. Find primitive root for 23.

→ $q = 23$ (is prime); $\alpha = ?$?

$\phi(23) = 22$; $\phi(\phi(23)) = \phi(22) = \phi(11 \times 2) = (11-1)*(2-1) = 10$?

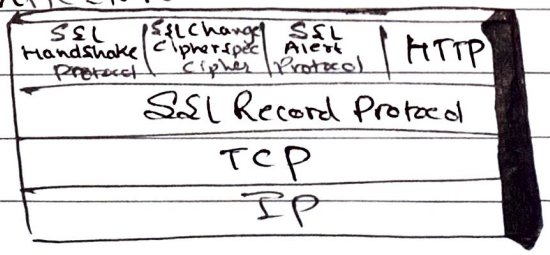
23/11/2022.

* Mandatory Class (Last Class)

SSL: Secure Socket Layer.

➢ Showed website's secure certificate.

• SSL Architecture.



SSL Handshake Protocol
SSL change Cipher Space Cipher
SSL Alert Protocol
HTTP
SSL Record Protocol
TCP
IP

- SSL Session : association between client and server, created by Handshake Protocol
- SSL Connection : transient, peer2peer communications link. associated with 1 SSL session.

• SSL Record Protocol:

→ Confidentiality
- Using symmetric encryption with a shared secret key defined by Handshake Protocol.

→ Message Integrity
- Using a MAC with shared secret key.
- Similar to HMAC but with different padding.

__/__/____

SSL CHANGE CIPHER SPEC PROTOCOL:

- one of 3 SSL specific protocols which use the SSL Record Protocol.
- a single msg. - causes pending state to become current.
- hence updating the cipher suite in use.

SSL ALERT PROTOCOL

- conveys SSL related alerts to peer entity
- severity : warning or fatal.
- Specific alert :
  - Compressed & encrypted like all SSL data.

Specific alert → unexpected msg, bad record mac, decompression failure, handshake failure, illegal parameter
→ close notify, no certificate, bad certificate, unsupported certificate, revoked certificate etc.

SSL HANDSHAKE PROTOCOL:

(diagrams.