10 security principles:

1) Economy of mechanism : Keeping security measures simple → effective

2) Fail-safe defaults:

keep thing safe, starting with strict rules

→ new users ~~don't~~ have minimal access rights to services

Final 3) Complete mediation

check permission access to resources every time when requested.

Ex: → bkash payment needs PIN to prevent unauthorized access

4) Open design : security design and architecture → public

cryptographic key → private

Pros: others can find and fix issues

5) Separation of privilege

→ multiple conditions should be fulfilled to achieve access to resources

→ 2 instead of 1 keys

6) Least privilege: program/ user get the minimum privilege that's necessary to complete the job.

7) Least common mechanism: → limit sharing

when multiple users access to a file

→ need separate channel to access

→ one's action doesn't affect other's security

8) Psychological acceptability: clean interface

fulfill expected security

9) ~~Work factor:~~ compromise recording:

not focus on stopping intruder

not using complex ways to stop them

– just watch and learn

10) ~~Compromise recording~~: work Factor

– think how much it cost someone to break security vs. what they've.

– students grade < millitary info

need more security

Access control has 4 mechanisms:

1) Access control matrices: A table that sets rules for who can do what with certain things.

col — file, folder each cell — what can do

row → person group system

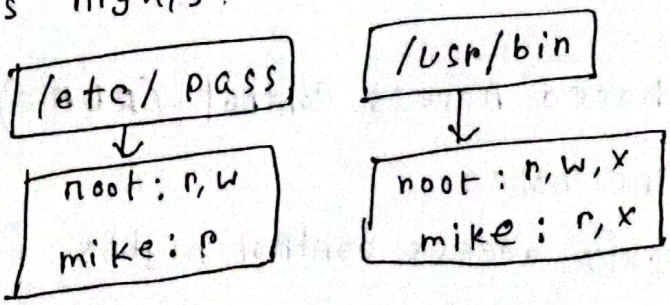
	/etc/pass	usr/bin
root	R, W	R, W, exec
mike	R	R exec

Pros: 1) easily see a cell and and tell which user can do what with a file

2) easy format

Cons: huge size, complex, hard to manage, time-consuming

2) Access control lists: Each object/file has a list of users who can access it and their access rights.



Pros: smaller, convinent to use

Cons: Finding out all the access rights for a user → slow

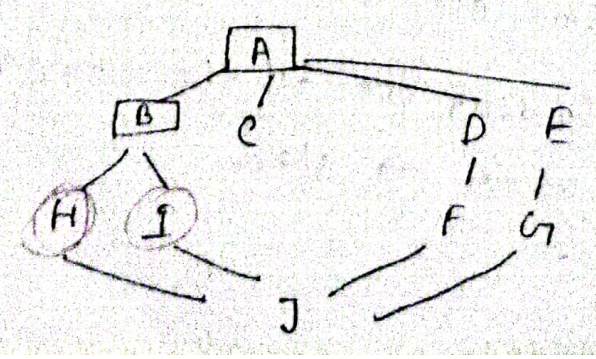
3) **Capabilities**: For each user, it lists the objects they can access and what they can do with each object.

root → | /etc/pass : r, w, x
| /admin : r, w

4) Pros :- Admins only need to handle access for pairs with actual rights
- reduce workload
- easy to find

Cons - complex,
- slow, because rights for a file are not directly linked

4) **Role based Access Control** (RBAC)
→ define roles
→ specify access control rights for these roles
→ not for subjects directly



B can inherit the access rights of H, I

Pros: 1) have fewer roles than subjects

2) easy to manage access rights

3) easy check to see if any user has rights.

Cons:

1) not commonly used in current OS

Final * Describe a process through which a system can fulfill the complete mediation security principal.

→ 1) when user access an object, they send a request to the system.

2) check authorization.

3) verify permission.

repeat for every attempt

Cost of circumventing: resource needed for attacker to defeat a security system

Security by obscurity: keep algorithms private

→ opposite of open design

Kerckhoff's principle

→ A cryptosystem should be secure even if everything about the system, except the key, is public knowledge

E

Encryption: A means to allow two panties to establish confidential communication over an insecure channel that is subject to eavesdropping.

plaintext (M) $\xrightarrow[\text{algo}]{\text{encryption}}$ Ciphertext (C)

$C = E(M)$, $M = D(C)$

D → Decryption

7 components of cryptosystem:

set of
1) plaintexts
2) ciphertexts
3) encryption keys
4) decryption "
5) correspondence between encryption and decryption keys
6) encryption algo
7) decryption "

Caesar cipher: one of the cryptosystem

→ shifting letter

$(K>0)$ encryption key, $\{K = 3\}$ AB → EF → forward shift

$(K<0)$ decryption key, $\{K = -3\}$ EF → AB → backward "

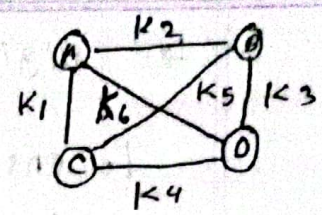
$S(C, K)$: Ex: $S(A, 3) = D$

letter key

$S(D, -3) = E$

sender and receiver has same key

Symmetric key distribution:



Pros : fast and efficient

Cons : each pair needs a separate secret key

* n parties need $n(n-1)/2$ keys

Public key cryptography:

B has 2 keys : public P_B
private S_B

A $\xrightarrow[\text{message}]{\text{send encrypted}}$ B , needs P_B , $\boxed{C = E_{P_B}(M)}$

B $\frac{\text{decrypt}}{\text{message}}$, needs S_B $\boxed{M = D_{S_B}(C)}$

* n person has 2n keys

Pros : provides strong security guarantees

Cons : slower than symmetric
unsuitable for interactive sessions
larger key length

Combining symmetric and public key system:

Pros: 1) Efficient and fast
2) shared secret key → ensures secured data transmission

Cons: complex, additional computational overhead

Digital signature: It's needed

To ensure that the text has come from a specific sender

~~$D_{SB}(E_{PB}(M)) = M$~~

$E_{PB}(D_{SB}(M)) = M$

→ reversal of encryption and decryption order

→ A signs the contract using secret key

→ B verifies the sign using A's public key

Pros:

- ensure messages are authentic and unchanged during transmission
- reliable and secured
- easy validation using sender's public key

Cons

- complex key mgmt.
- signature size can be large as message
- slow performance
- transmission and storage cost is high.

Crypto-system 2 attacks:

1) MITM (Man in the middle): It occurs when an unauthorized 3rd party interrupts communication between 2 parties.

3nd Panty can — add, delete, modify, view traffic
— potentially compnomise confidentiality
authenticity
integnity of
message

s → signature

* Bob send $(m, s) : (c, s)$
* 3nd penson change (c', s')
* A thinks it's from B → $(c', s') : (m', s')$

3nd penson can't decnypt and see the real message

2) **Bnute-fonce attack**: tnying diffenent pnobable (possible) keys oven a ciphentext to decnypt it into meaningful text. (tny eveny possible keys)

If the neal message (m) is
→ binany stning, hand to find the valid message
→ if natunal language, find meaningful nesults aften a few tny.

* **Unicity distance**: It's the point at which the ciphentext becomes long enough fon an attacken to successfully decnypt it, necoven the onignal message, even without knowing the key.

$$n = \frac{K}{1-\alpha}$$

k – key length
α – constant

** Cons: 1) ineffective for long, random, complex passwords
2) more time, resource, power needed

---

* English text: 1 character = 8 bits

Suppose, $n = 8t$ $\quad t$ " $= 8t$ "

$\rightarrow t = \frac{n}{8}$ $\quad \therefore$ total possible array $= 2^{8t} = 2^n$

1 char $\rightarrow$ 1.25 bits info

$\therefore t$ " $\rightarrow 1.25t$ bits

$\therefore$ no. of t-byte array $= 2^{1.25t}$

$= 2^{1.25n/8} = 2^{0.16n}$

valid text $= 2^{\alpha n}$ [α - constant]

$\therefore$ probability of getting valid text

$$= \frac{2^{\alpha n}}{2^n} = 2^{(\alpha-1)n} = \frac{1}{2^{(1-\alpha)n}}$$

Now, $k \rightarrow$ length of decryption key

for a ciphertext $\xrightarrow{k \text{ keys}}$ 2^k plaintext

$\therefore$ probabality of getting plain text

$$= \frac{2^k}{2^{(1-\alpha)n}}$$

Final 17

1) ASCII → 1.25 bits info → 32 bit sequence

Calculate the probability of getting meaningful text message

→ probability valid text $= \frac{1}{2^{(1-\alpha)n}} = \frac{1}{2^{(1-1.25)32}} = 256$

∴ probability (among all combo) $= \boxed{\frac{256}{2^{32}}}$ (Ans)

Final

2) 1.25 bits info per character.

8 bit ASCII

6.75 bits per character are redundant.

→ probability $= \frac{\text{total valid msg}}{\text{no. of total } t \text{ byte msg}} = \frac{2^{1.25t}}{2^{8t}}$

$= 2^{-6.75t}$

How can MAC ensure data integrity?

Ch-5

MAC

1) What is message Authentication Code?

→ In cryptography, MAC is a short piece of information used for authenticating and integrity-checking a message. It ensures that the message is coming from the correct sender, has not been changed, the data transferred over a network is legitimate and doesn't contain harmful code.

Ex: ① Message creation:

— Alice's message: "Hello, Bob!"

② MAC Generation:

- Alice uses a secret key (shared with Bob) and a MAC algorithm to generate a MAC for the message.
- suppose secret key is "secret123"
- " algo is "HMAC-SHA256"
- The MAC is "5d4140..."

③ Message transmission:

Alice sends message + MAC

" Hello, Bob! " + " 5d4140 . . . "

4) MAC Verification:

- Bob neciere msg + MAC
- genenate MAC using same seenet key and MAC algo
- if (received MAC == Bob's generated MAC)
  - msg is from Alice
  - msg is not changed
- else
  - not from Alice / altened / modified

* What's Dictionary Attack?

→ a method used by attackers to guess passwords with a dictionary list of common words / phrases used by businesses and individuals.

→ a type of brute force attack

→ trying out every possible word in dictionary

* What is social engineering attack?

→ tactic of manipulating, influencing or deceiving a victim in order to gain control over a computer system or to steal personal or financial information. It uses psychological manipulation to trick users into making security mistakes or giving away sensitive information.

* What is pretexting attack?

→ use of a fabricated story to gain a victim's trust and trick or manipulate them into sharing sensitive information, downloading malware, sending money to criminals or otherwise harming themselves or the organization they work for.

* How Digital Certificate works?

→ Digital certificates verify identities and enable secure, encrypted communication.

CamScanner

Steps: (i) A trusted Certificate Authority (CA) issues a digital certificate after verifying the entity's identity.

(ii) The entity installs the certificate on its server.

(iii) The server presents the certificate to user's browser.

(iv) The user's browser verifies the certificate.

(v) If valid, a secured, encrypted connection is established.

* What is the $\boxed{\text{role of CA}}$ (Certificate Authority)

→ CA is a trusted organization that issues digital certificates.

Role: 1) verifies identity of entities

(2) creates and signs digital certificate,

(3) Enable secure communication between users and browsers.

CamScanner

→ something for something → Latin word

* Quid Pro Quo Attack:

→ is a type of social engineering attack in which the attacker promises the victim a favor in exchange for information or other benefits.

Ch-8

* GCD (2260, 812) using Euclidean Algo:

→ ① $a = 2260$, $b = 812$

$\therefore a \div b = 2$, rem $= 636$

② $a = 812$, $b = 636$

~~812 %~~ $a \div b = 1$, rem $= 176$

③ $a = 636$, $b = 176$

$a \div b = 3$, rem $= 108$

④ $a = 176$, $b = 108$

$a \div b = 1$, rem $= 68$

⑤ $a = 108$, $b = 68$

$a \div b = 1$, rem $= 40$

CamScanner

* AES — Advanced Encryption standard

(6) a = 68, b = 40, division = 1, rem = 28

(7) a = 40, b = 28, div = 1, rem = 12

(8) a = 28, b = 12, div = 2, rem = 4

(9) a = 12, b = 4, div = 3, rem = 0

∴ rem, = 0, so the GCD is b ~~div~~ = 4

* GCD (226, 12)

→ (i) a = 226, b = 12, div = 18, rem = 10

(ii) a = 12, b = 10, div = 1, rem = 2

(iii) a = 10, b = 2, div = 5, rem = 0

ans

* 5^{31} mod 13 using repeated squaring:

→ $31 = 16 + 8 + 4 + 2 + 1$

$5^{31} = 5^{16+8+4+2+1}$

$= 5^{16} \cdot 5^{8} \cdot 5^{4} \cdot 5^{2} \cdot 5^{1}$

$= (8 \times 12 \times 8 \times 12 \times 5) \bmod 13$

$= 7680 \bmod 13 = 8$

$5^{1} \bmod 13 = 5$

$5^{2} \bmod 13 = 12$

$5^{4} \bmod 13 =$

$(12 \times 12) \bmod 13$

$= 144 \% 13 = 8$

$5^{8} \bmod 13$

$= (8 \times 8) \% 13$

$= 12$

$5^{16} \bmod 13$

$= (12 \times 12) \bmod 13 = 8$

* Dexter wants to set up his own public and private keys. He chooses $p = 23$, $q = 19$ with $e = 283$. Find d so that ed has a remainder of 1 when divided by $(p-1)(q-1)$

$\rightarrow m = (p-1)(q-1) = 22 \times 18 = 396$

$ed = 283d$, rem $= 1$, when divided by $m = 396$

d	ed	rem (div by 396)
1	283	283
2	566	170
3	849	57
4	1132	340
5	1415	227
6	1698	114
7	1981	1

$\therefore$ for $d = 7$, $ed = 283 \times 7 = 1981$ has a rem of 1 when div by 396

* What's cryptanalysis?

→ study and process of analyzing and decrypting ciphers, codes and encrypted text without using the real key.

→ analyze cryptographic system

→ understand/ weakness and vulnerabilities identify.

* (I) Divide the plaintext into blocks of size $m = 3$.

Block 1 : BBC Block 2 : ABC

Block 3 : BCA Block 4 : A

(II) After padding, Block 4 : A22

(III) multiply each block by encryption key matrix :

$$K = \begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix}$$

B 1 :

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 9 \\ 25 \\ 41 \end{bmatrix} \rightarrow \begin{matrix} J \\ Y \\ P \end{matrix}$$

→ 41 mod 26 = 15

B2:

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ 20 \\ 32 \end{bmatrix} \rightarrow \begin{matrix} I \\ U \\ G \end{matrix}$$

B3:

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 5 \\ 17 \\ 29 \end{bmatrix} \rightarrow \begin{matrix} F \\ R \\ D \end{matrix}$$

B4:

$$\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \times \begin{bmatrix} 0 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} 15 \\ 39 \\ 63 \end{bmatrix} \rightarrow \begin{matrix} P \\ N \\ L \end{matrix}$$

$\therefore$ Encrypted text: JYP IUG FRD PNL

U - 6

Ciphen Block chaining

How to detect whethen CBC can hide pattenns on not ? Describe the process.

→ ① Encrypt a plaintext with repeating pattenns using CBC mode with a symmetric key and a random initialization vecton.

② Break the reselting ciphentext into blocks.

③ compane the ciphentext blocks fon any nep etitions.

If all ciphentext blocks are unique, CBC mode successfully hides the pattenns. If any blocks are identical, CBC mode fails to hide the pattenns effectively.

Between CFB and OFB, which has a betten encryption architecture ? why ?

→

CFB → Ciphen Feedback	OFB → Output Feedback
1) Ciphentext feedback.	1) Key stream generated from initialization vecton.
2) Ennon pnopagates fon only one block.	2) No ennon pnopagation.
→ If ennon in 1 block, it'll affect next block.	→ If ennon in 1 block, it'll affect only that specific block.

CamScanner

So OFB better

CFB	OFB
3) Slightly slower performance → due to feedback loop	3) Faster as the key stream can be pre-computed.
4) Encryption can't be parallelized. Decryption can be	4) Both encryption and decryption can be parallelized.

Q Hill cipher use and find cipher text.

string = dog

$$\vec{x} = \begin{bmatrix} 3 \\ 14 \\ 6 \end{bmatrix}$$

Given, random matrix

$$K = \begin{bmatrix} 1 & 2 & 5 \\ 3 & 3 & 4 \\ 2 & 1 & 3 \end{bmatrix}$$

$$\vec{c} = K \cdot \vec{x} = \begin{bmatrix} 1 & 2 & 5 \\ 3 & 3 & 4 \\ 2 & 1 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 14 \\ 6 \end{bmatrix}$$

$$= \begin{bmatrix} 3+28+30 \\ 9+42+24 \\ 6+14+18 \end{bmatrix} = \begin{bmatrix} 61 \\ 75 \\ 38 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 9 \\ 23 \\ 12 \end{bmatrix}$$

∴ cipher text = ~~h~~ J x m

Electronic codebook

Why is ECB bad with images?

→ bad for encrypting images because it handles each block of data separately. This means that identical parts of the image will look the same even after encryption, making patterns in the original image visible in the encrypted version. As a result, repetitive structures and textures can still be seen, compromising security. (CBC is better) hides pattern

Suppose the number of keys for substitution cipher is 26!. If we partition the plaintext into bignam, what'll be the no. of keys in the keyspace?

→ Total possible bignams = 26 × 26 = 676

pair (AA, AB ... ZZ etc)

∴ Total keys = 676!

In case of trignam (26³)!

Disadvantages of substitution cipher:

1) It doesn't hide how often letters appear, so someone could guess which letters stand for which.

CamScanner

2) There are only a limited number of keys. making it easier to try them all and decode the message.

3) Doesn't hide patter.

4) Not secured.

5) It's hard to keep the keys secret and make sure only the right people can decode the message.

What's One-time pad?

→ one type of substitution cipher that is absolutely unbreakable.

→ uses a random key as long as the message for encryption, ensuring perfect secrecy through XOR operations

What is block cipher?

→ a cryptographic algorithm that encrypts fixed-size blocks of plaintext into ciphertext.

CamScanner

Briefly explain ECB and CFB modes of block ciphers.

→

ECB	CFB
1) encrypts each block of plaintext independently with the same key.	1) encrypts a plaintext block by XORing it with the output of the encryption of the previous ciphertext block (or initialization vector for the 1st block)
2) Simple and straightforward. Faster.	2) Slower than ECB
3) Errors in 1 block does not affect others.	3) Affects subsequent blocks until synchronization is restored.
4) Parallelizable	4) Not parallelizable.
5) Reveals pattern	5) hides pattern

(*) text = TOM, find cipher using hill cipher.

$$k = \begin{bmatrix} 1 & 2 & 5 \\ 3 & 3 & 4 \\ 2 & 1 & 3 \end{bmatrix}, \quad x = \begin{bmatrix} 19 \\ 14 \\ 12 \end{bmatrix}$$

$$\therefore c = \begin{bmatrix} 19+28+60 \\ 57+42+48 \\ 38+14+36 \end{bmatrix} = \begin{bmatrix} 107 \\ 147 \\ 88 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 3 \\ 17 \\ 10 \end{bmatrix}.$$

$\therefore$ cipher text = d r k

* Encrypt the text "CAB BCC ACC B" using block cipher with padding, $m = 3$

$$K = \begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix}$$

for the 1st block, $\vec{x} = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$

for " last " , $\vec{x}' = \begin{bmatrix} 1 \\ 3 \\ 3 \end{bmatrix}$

→ After padding $B_1 = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$

$B_2 = \begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}$, $B_3 = \begin{bmatrix} 0 \\ 2 \\ 2 \end{bmatrix}$, $B_4 = \begin{bmatrix} 1 \\ 3 \\ 3 \end{bmatrix}$

$$K \cdot B_1 = \begin{bmatrix} 2+3 \\ 10+7 \\ 18+11 \end{bmatrix} = \begin{bmatrix} 5 \\ 17 \\ 29 \end{bmatrix} \bmod 26 = \begin{bmatrix} 5 \\ 17 \\ 3 \end{bmatrix} = \text{frd}$$

$$K \cdot B_2 = \begin{bmatrix} 1+4+6 \\ 5+12+14 \\ 9+20+22 \end{bmatrix} = \begin{bmatrix} 11 \\ 31 \\ 51 \end{bmatrix} \bmod 26 = \begin{bmatrix} 11 \\ 5 \\ 25 \end{bmatrix} = \text{Lfz}$$

$$K \cdot B_3 = \begin{bmatrix} 4+6 \\ 12+14 \\ 20+22 \end{bmatrix} = \begin{bmatrix} 10 \\ 26 \\ 42 \end{bmatrix} \bmod 26 = \begin{bmatrix} 10 \\ 0 \\ 16 \end{bmatrix} = \text{kaq}$$

$$K \cdot B_4 = \begin{bmatrix} 1+6+9 \\ 5+18+21 \\ 9+30+33 \end{bmatrix} = \begin{bmatrix} 16 \\ 44 \\ 72 \end{bmatrix} \bmod 26 = \begin{bmatrix} 16 \\ 18 \\ 20 \end{bmatrix} = \text{qsu}$$

CamScanner

Problem and solution of block cipher with padding mechanism:

→ can accidentally leak information about the length of the message on even parts of the message itself. To fix this, use standardized padding methods, like PKCS# 7, and double-check for mistakes during decryption.

→ encrypt it separately to protect length.

→ Always choose padding that adds the least amount of data possible.

ECB Problem: leaks patterns, reveal info about plaintext, not suitable for sensitive info.

→ identical plaintext blocks are encrypted into identical ciphertext blocks.

CFB problem: ① more complex, slower.

② Initialization vector is needed.

③ If error occurs in a block of ciphertext, it affects the subsequent block's decryption untill syn-chronization is restored, data corruption occurs

④ not suitable for large-scale data

CamScanner

Why is padding used in encryption?

→ to ensure that plaintext message fills up the entire block size required by the encryption algorithm.

→ make efficient

→ prevent leakage of info

4 modes of block-cipher:

1. ECB (Electronic Codebook)
2. CBC (Cipher-block chaining)
3. CFB (Cipher feedback)
4. OFB (output feedback)

Vigenère cipher uses a key that is as long as plain text, T or F?

→ False. It uses a key that is repeated to match the length of plaintext.

AA's security test

1) verify encryption method used for pic

2) Test decryption 3) Test pin's strength

4) check access control

CamScanner

Q)

Encryption	Hashing
① converts plaintext into ciphertext using a key.	① generates a fixed size hash value from input data.
② Allows decryption with the same key to retrieve the original plaintext.	② Hash values can't be reversed to obtain the original input.
③ Ensures only authorized parties can access and understand the data.	③ used for data validation, digital signatures and securely storing passwords.

Q) In CBC mode we can take advantage of pre computed IV. T or F?

→ False. CBC needs an IV that's random and unique for each encryption operation. It's not pre-computed or reused.

Q) What are problems of security questions in password reset systems? Example.

easily guessable

vulnerable system → ① often predictable ② Limited security ③ often unchanged answers

Ex: 2008, Sarah Palin's Yahoo account hacked.

CamScanner

The attacken used publicly available info to answen security questions

→ neset password

→ gain unauthonized access to private mails

text = CBB BAC ABC B

After padding, $B_1 = \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix}$, $B_2 = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$

$B_3 = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$ $B_4 = \begin{bmatrix} 1 \\ 3 \\ 3 \end{bmatrix}$

$$K.B_1 = \begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \\ 9 & 10 & 11 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2+2+3 \\ 10+6+7 \\ 18+10+11 \end{bmatrix} = \begin{pmatrix} 7 \\ 23 \\ 39 \end{pmatrix} \bmod 26$$

$$= \begin{pmatrix} 7 \\ 23 \\ 13 \end{pmatrix} = hxn$$

$$K.B_2 = \begin{bmatrix} 1+6 \\ 5+14 \\ 9+22 \end{bmatrix} = \begin{bmatrix} 7 \\ 19 \\ 31 \end{bmatrix} \bmod 26 = \begin{bmatrix} 7 \\ 19 \\ 5 \end{bmatrix} = htf$$

$$K.B_3 = \begin{bmatrix} 2+6 \\ 6+14 \\ 10+22 \end{bmatrix} = \begin{pmatrix} 8 \\ 20 \\ 32 \end{pmatrix} \bmod 26 = \begin{bmatrix} 8 \\ 20 \\ 12 \end{bmatrix} = ium$$

$$K.B_4 = \begin{bmatrix} 1+6+9 \\ 5+18+21 \\ 9+30+33 \end{bmatrix} = \begin{pmatrix} 16 \\ 44 \\ 72 \end{pmatrix} \bmod 26 = \begin{bmatrix} 16 \\ 18 \\ 20 \end{bmatrix} = qsu$$

CamScanner

Ron Rivest
Adi Shamir
Leonard Adleman

RSA Cryptosystem: a widely used public key cryptographic system that facilitates secure data transmission.

key components:

1) public key : $\{n, e\}$
2) private key : $\{d, n\}$

Math Step:

1) Given, p, q (large prime numbers)
2) $n = p \times q$
3) $\phi(n) = (p-1)(q-1)$
4) choose e such that
   a) $1 < e < \phi(n)$
   b) $\gcd(e, \phi(n)) = 1$
5) find d, $d \times e \equiv 1 \pmod{\phi(n)}$
6) Encryption, $C = M^e \bmod (n)$ → use public key
7) Decryption, $m = C^d \bmod n$. → use private key

Example: Let, $p = 61$, $q = 53$

$n = 61 \times 53 = 3233$

CamScanner

$\phi(n) = (61-1)(53-1) = 3120$

Suppose, $e = 17$

$\because \gcd(17, 3120) = 1$

$d = 17^{-1} \bmod 3233$

$= 2092$

Let $m = 65$. $\quad c = 65^{17} \bmod 3233$ —— ①

$m = (65^{17})^{2092} \bmod 3233$ —— ②

$17 = 8 + 4 + 2 + 2 + 1$

① $\rightarrow \therefore \; 65^{17} = (65^8 \cdot 65^4 \cdot 65^2 \cdot 65^2 \cdot 65^1) \bmod 3233$

$= (1547 \times 1232 \times 992^2 \times 65) \bmod 3233$

$= 2790$

$65^1 \bmod 3233$
$= 65$

$65^2 \bmod n$
$= 4225 \bmod n$
$= 992$

$65^4 \bmod n$
$= (992 \times 992) \bmod n$
$= 1232$

$65^8 \bmod n$
$= (2237)^2 \bmod n$
$= 1547$

$\therefore c = 2790$

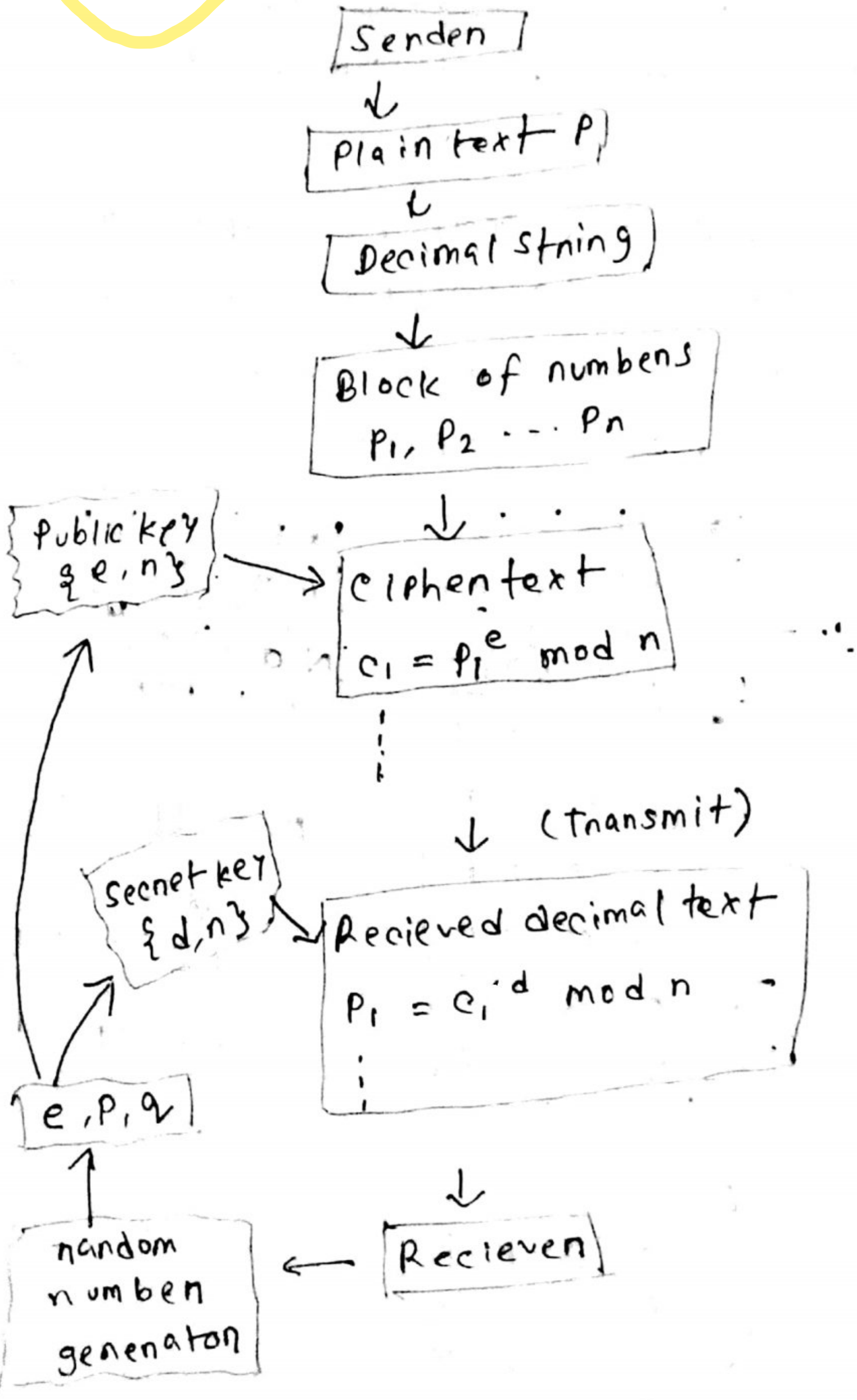
$m' = 2790^{2092} \bmod 3233$

$= 2699$

$\therefore m \neq m'$

CamScanner

* General pipeline :



CamScanner

* Play fain ciphen :

- symmetric encryption
- dignam substitution ciphen
  ↓
  2

ques TT

* Suppose a group of people agreed upon a key, k = "PUZZLE". now find the ciphentext of the plaintext, m = "LITTLE" using playfain ciphen.

→

k =

P	U	Z	L	E
A	B	C	D	F
G	H	I/J	K	M
N	O	Q	R	S
T	V	W	X	Y

m = LITTLE

= LI TX TL EX

ZK VY XP LY

CS CamScanner

CS CamScanner

* Diffie - Hellman key exchange protocol:

→ 2 person share a value on public channel $\{p, g\}$

p → large prime

g → base

→ each person chooses secret number (a/b)

→ compute public keys value:

$g^a \bmod p$; $g^b \bmod p$

→ exchange public value

→ computed shared secret:

$(g^a)^b \bmod p == (g^b)^a \bmod p$.

Pros of IPsec (Internet Protocol Security)

→ Internet secure at IP layer

(1) provides strong encryption and authentication to protect data

(2) Ensures data is not altered.

(3) works well with existing net/app and protocols without modification

(4) flexible and Transparent

CamScanner

* **Playfain:** K = ASGARD

M = GROOT

GR OX OT

RD QV PU

K =

A	S	G	R	D
B	C	E	F	H
I/J	K	L	M	N
O	P	Q	T	U
V	W	X	Y	Z

* **Elgamal Cryptosystem:**

Given, $p = 7$, $g = 3$

(a) ① select private key. $1 \le S_k \le p-2$

$1 \le S_k \le 7-2$

$\therefore \boxed{S_k = 5}$

② $P_k = g^{S_k} \bmod p$

$y = 3^5 \bmod 7$

$= 243 \bmod 7 = 5$

Ans: $\therefore$ public key $(p, g, y) = (7, 3, 5)$

private key, $x = 5$

CamScanner

(b) $P_k : (p = 7, g = 3, y = 5)$, $m = 13$

random k : $1 \le k \le p-2$ $\therefore k = 4$

$c_1 = g^k \bmod p = 3^4 \bmod 7 = 4$

$c_2 = m \cdot y^k \bmod p = 13 \cdot 5^4 \bmod 7 = 5$

$\therefore (c_1, c_2) = (4, 5)$

* Why are RAIDs used?

→ Redundant Array of Independent Disks.

→ data is stored in multiple disks. If 1 disk fails, no data loss.

→ data reliability and availability through redundancy and fault tolerance

→ distribute data across multiple disks

What is SSH?

Ans: SSH (Secure Shell) is a network protocol that allows secure access to remote computers or servers over an encrypted connection. It is commonly used for managing servers, transferring files, and executing commands remotely while ensuring data confidentiality and integrity.

CamScanner