Encryption: A means to allow two parties 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 "

Caesan 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

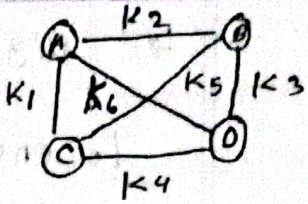
$S(C, K)$: Ex: $S(A, 3) = D$

↓ letter ↓ key

$S(D, -3) = E$

→ senden and neceiven has same key

Symmetnic key distnibution :



pnos : fast and efficient

cons : each pain needs a separate secnet key

* n panties need n(n-1)/2 keys

Public key onyptognaphy :

B has 2 keys : public P_B
pnivate S_B

A —(encnypted message, send)→ B , needs P_B , $C = E_{P_B}(m)$

B —(decnypt message)→ , needs S_B $M = D_{S_B}(C)$

* n penson has 2n keys

pnos : pnovides stnong secunity guanantees

cons : slowen than symmetnic
unsuitable fon intenactive sessions
langen key length

Combining symmetnic and public key system :

pnos : 1) Efficient and fast
2) shaned secnet key → ensunes secuned data tnansmission

Cons : complex, additional computational ovenhead

Digital signature: It's needed to ensure that the text has come from a specific senden

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

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

→ 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 senden'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 intercepts communication between 2 parties.

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

s → signature

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

3rd penson can't decrypt 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, $n = 8t$ $\quad t \quad u \quad = 8t \; u$

$\rightarrow t = \frac{n}{8}$ $\quad \therefore$ 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 $= 2^{1.25t}$

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

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

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

$= 2^{-6.75t}$