Date _____ Day _____

▷ PUBLIC-KEY ENCRYPTION

→ Uses mathematical functions (eg: modular arithmetic, exponentiation)

→ It is asymmetric encryption
- ↳ Public key (shared openly)
- ↳ Private key (kept secret)
- ↳ This dual-key design impacts confidentiality, key distribution and authentication

• Common Misconceptions

#01 ⇒ Public-key encryption is more secure than symmetric encryption
- ↳ FALSE: Security depends on key length & computational efforts to break the cipher not on whether it's symmetric or asymmetric

#02 ⇒ Public key encryption makes symmetric encryption obselete
- ↳ FALSE: Public-key algo are much slower Sym Encry. (fast) still used, often combined with pub-key encr. for key exchange.

Date _____ Day _____

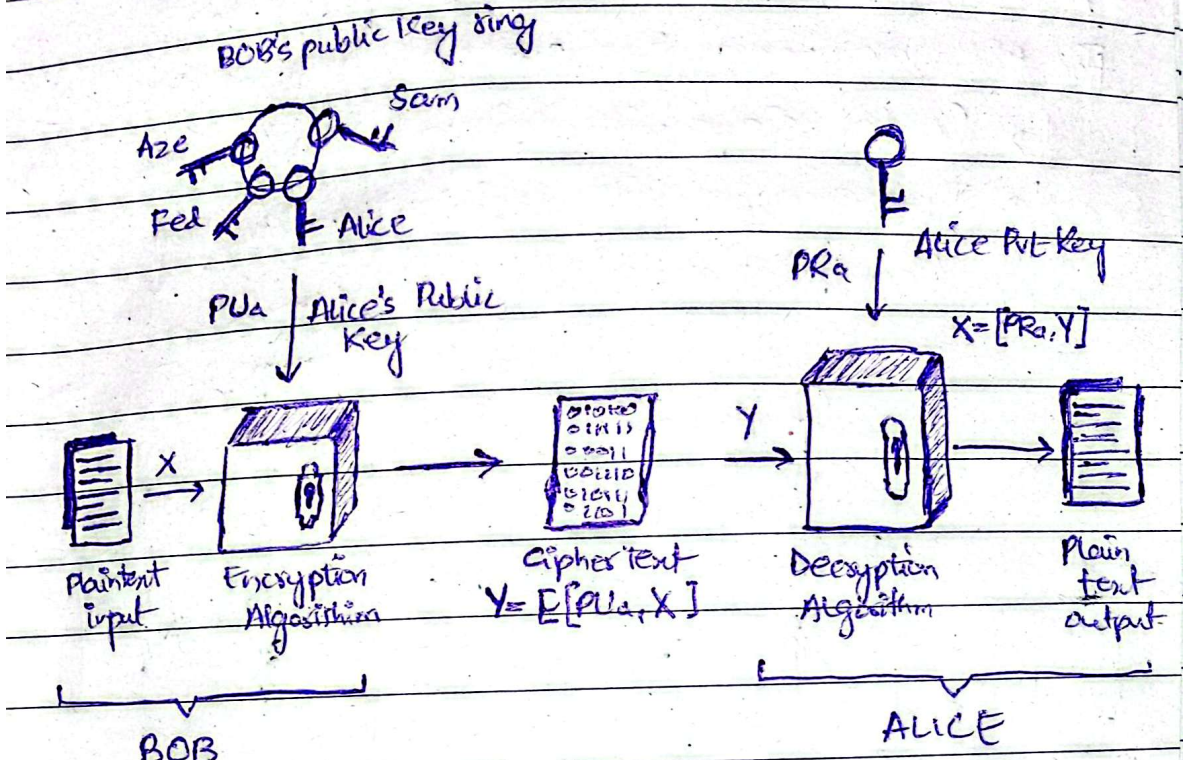
- Ingredients of Public key
  - ↳ Plain Text
  - ↳ Encryption Algo
  - ↳ Pub & Pvt key
    - ↳ for encrypp
    - ↳ for decrypp
  - ↳ Cipher Text
  - ↳ Decryption Algo

- Steps
  1. Each user generates a key pair (public key + private key)
  2. The public key is shared openly; the pvt key is kept secret
  3. To send a secure message, the sender encrypts it with the recipients public key
  4. The recipient decrypts it with their private key, ensuring only they can read it.

- Ensuring Confidentiality
  - → Sender uses recipient's public key → only recipient (with private key) can decrypt that msg no one else, therefore confidentiality is maintained as long as pvt key is kept safe.

Date _____ Day _____

→ Confidentiality depends on
  ↳ Strength of the encryption algorithm
  ↳ Protection of the private key
  ↳ Security of supporting protocals.



▲ Encryption with Public Key.

• Ensuring Authentication → Since only Bob has it's private key so data integrity also preserved

→ Sender uses their private key to encrypt a message. [only sender has access to it's pvt key]

→ Receiver or anyone uses the sender's public key to decrypt / verify.

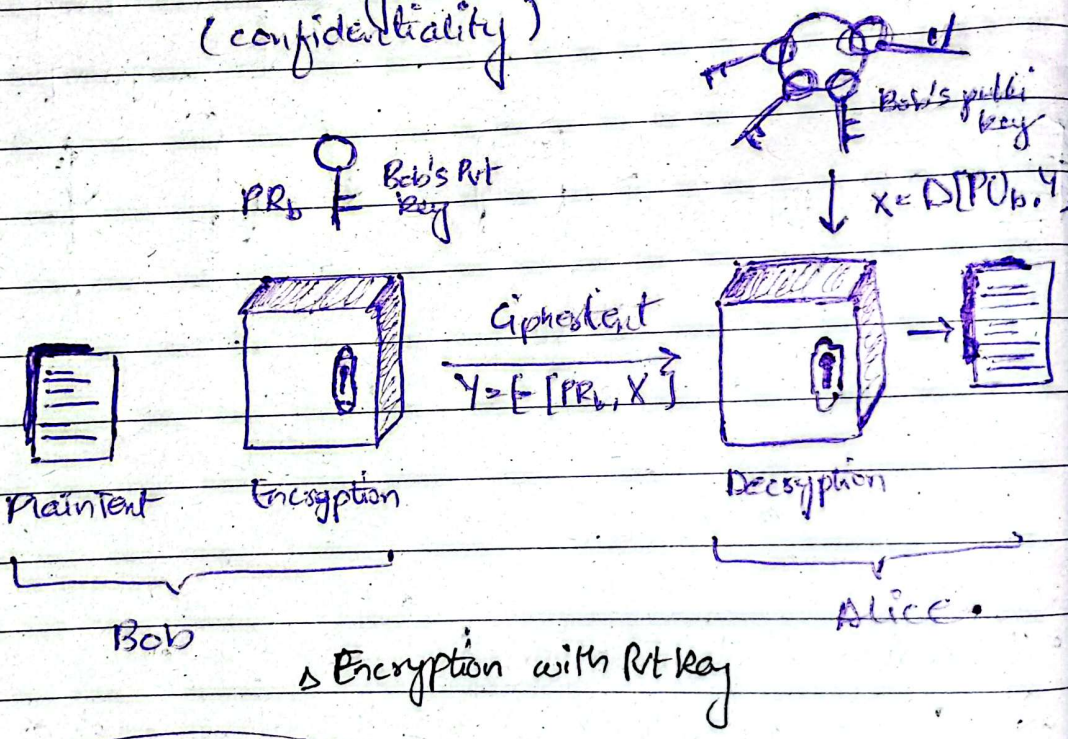
↳ If decryption works, it proves the message came from the sender bcz only the message decrypted from the ... can be decrypted thru Alice's public key.

Day

Date

→ Therefore authentication is achieved

NOTE: Hheese only authentication is preserved not confidentiality bc2 anyone with Alice's public key can view the msg.
For this we need to combine both keys
↳ Sender signs with their pvt key (authentication)
↳ Then encrypts with recipients public key (confidentiality)



∴ Encryption with Pvt Key

MAIN DIFFERENCE

① Confidentiality:
↳ Encryption ⇒ Pub key
↳ Decryption ⇒ Pvt key

② Authentication
↳ Encryption ⇒ Pvt Key
↳ Decryption ⇒ Pub Key

Date _____ Day _____

Requirements Of Public Key Cryptography

(1) It is computationally easy for a party B to generate a pair

(2) It is computationally easy for a sender A, knowing the public key & msg to be encrypted M, to generate the corresponding CT

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

(3) It is computationally easy for the receiver B to decrypt the resulting CT using the Pvt Key to recover the original msg

$$M = D(PR_b, C) = D[PR_b, E(PU_b, M)]$$

(4) It is computationally infeasible for an opponent, knowing the pub key, PU_b, to determine Pvt key, PR_b

(5) It is computationally infeasible for an opponent knowing pub key, PU_b and a ciphertext "C", to recover original message, M.

Date _____ Day _____

- RSA

→ First major public key encryption scheme
→ A block cipher where plain text & cipher text are integers b/w 0 — n-1 for some n
→ 1024-bit key (300 decimal digits) is consider a strong key for most applications

See Table #2.3

▷ DIGITAL SIGNATURES AND KEY MANAGEMENT :

- Digital Signatures

→ A digital signature is the result of a cryptograph transformation of data
→ It provides
① Origin Authentication — Confirms who sent the data
② Data Integrity — Ensures the data hasn't been altered
③ Non-repudiation — The sender cannot deny having signed the message

A digital signature is a unique bit pattern generated from
↳ The data ↳ The sender's pvt key

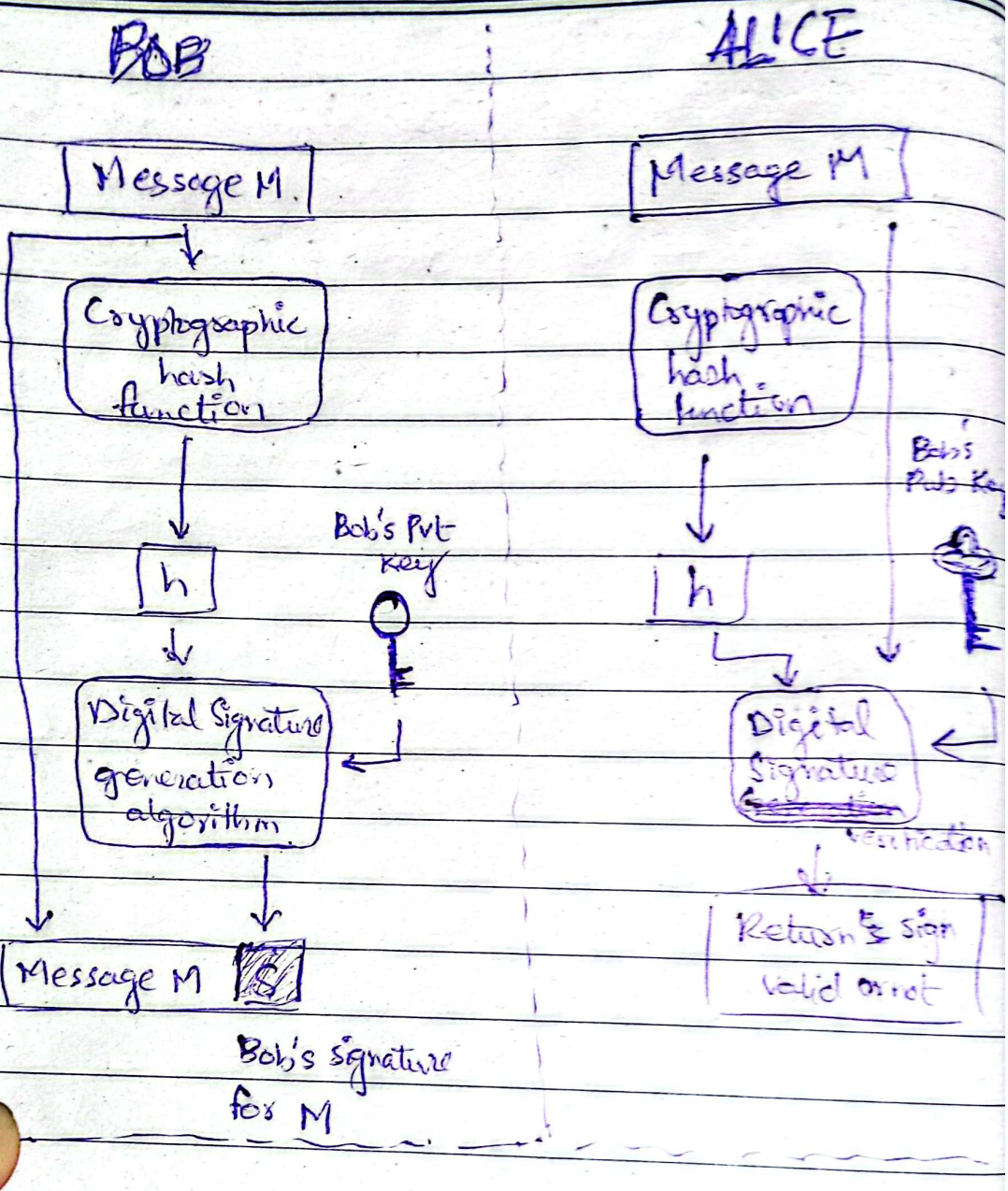
Date _____ Day _____

- Verification confirms
  1. The message was signed by the claimed signer
  2. The message has not been altered after signing
  3. The signer cannot later repudiate (deny) the signature.

→ Types of Digital Signature Algorithms
1. Digital Signature Algo (DSA)
2. RSA Digital " " (RSA)
3. Elliptic Curve Digital Signature Algo (ECDSA)

- Process
  - ▸ Signing Process
    1. Bob uses a secure hash function to generate a hash from a msg
    2. Key is generated thru hash + Bob's pvt key to produced a digital signature
    3. Bob sends msg + digital signature to Alice
  - ▸ Verification Process
    1. Alice computes hash value of msg
    2. Generates signature using hash + Bob's Public key
    3. Match the generated signature with the received one.

Date _____ Day _____



- Limitations
  - ↳ No confidentiality
    - ↳ The msg not encrypted
    - ↳ Even msg encrypted with pvt key can be decrypted anyone by public key
  - ↳ ~~Digital signature pvt~~

Date _____ Day _____

o Public Key Certificates

- **Problem:**
  - ↳ Public Keys are meant to be publically available
  - ↳ But there is a security risk:
    - ↳ Anyone can make a key & pretend to be Bob
    - ↳ The attacker can:
      - ↳ Read all encrypted message intended for Bob
      - ↳ Impersonate Bob in authentication
  - ↳ Detection of this vulnerability may take time!

- **Solution:**
  - ↳ A certificate = Public key + userID of the key owner (additional info: identity of CA, validation period)
  - ↳ The entire certificate is digitally signed by a trusted third party (Certificate Authority, CA)
  - ↳ Ensures public key belongs to the claimed user
    - Could be gov agency, bank or trusted security company

Date _____ Day _____

- Process

(1) User software (client) generates a key pair (pub + pvt key)

(2) Unsigned certificate creation — contains userID + user's public key

(3) User provides unsigned certificate to CA in a secure ~~manner~~ manner

(4) CA creates digital signature
   (a) Computes hash of unsigned cert &
   (b) CA signs this hash using CA's pvt key

(5) CA attaches signature with unsigned certificate to make it "Signed Certificate"

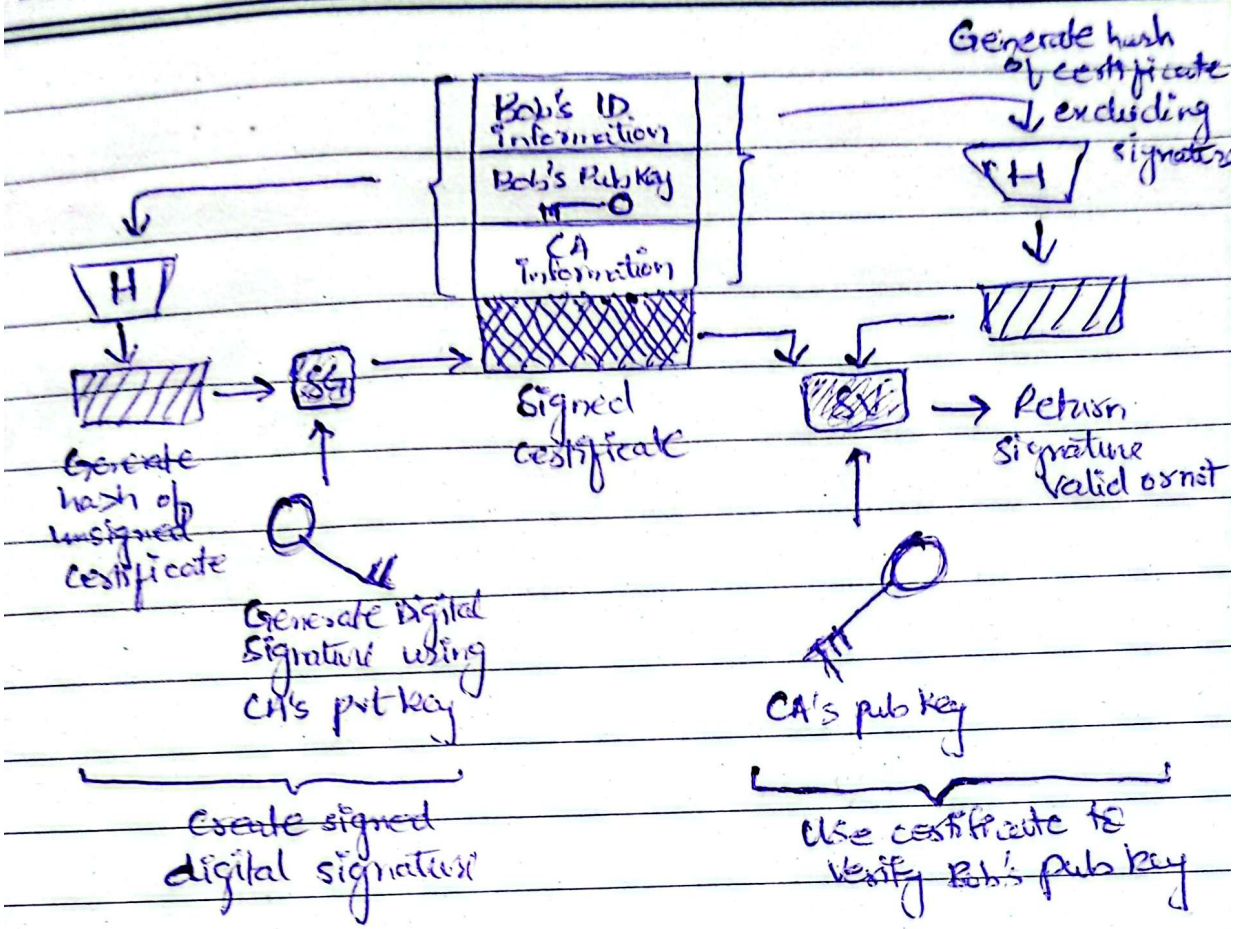
(6) CA returns signed certificate to user

(7) User now publishes the signed certificate not his public key.

- Verification by others
  ↳ Recver calculates hash of certificate (excluding signature)
  ↳ Recv verifies CA's signature using CA's pub key

- Application
  - IPsec
  - TLS/SSL
  - SSH (Secure Shell)
  - S/MIME (Secure email)

Date _____ Day _____


Bob's ID Information
Bob's Pub Key
CA Information
Signed Certificate
H
Generate hash of unsigned certificate
Generate Digital Signature using CA's pvt key
Create signed digital signature
Generate hash of certificate excluding signature
CA's pub key
Use certificate to verify Bob's pub key
Return signature valid or not