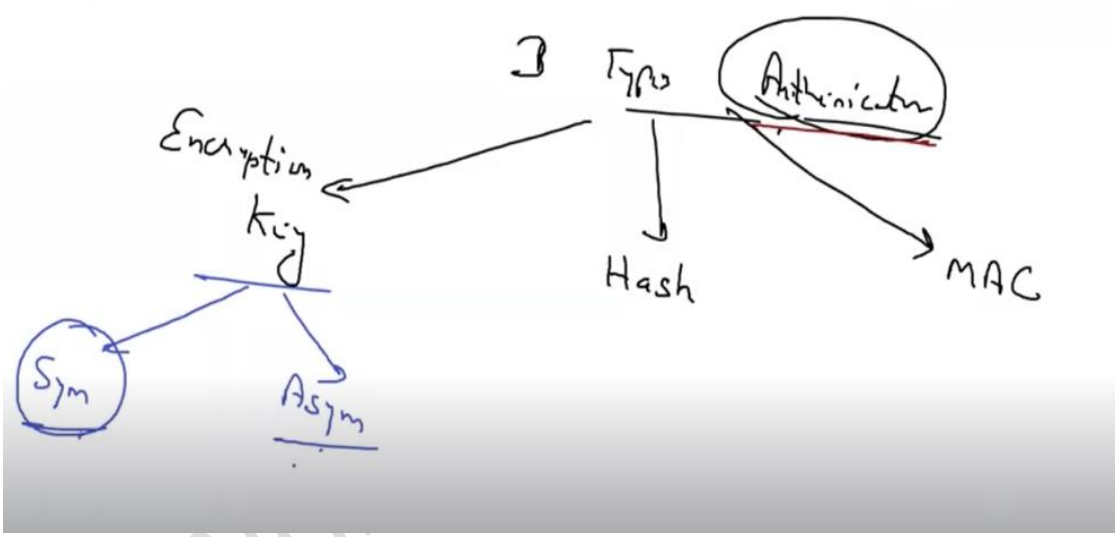
Cryptography Session No.5

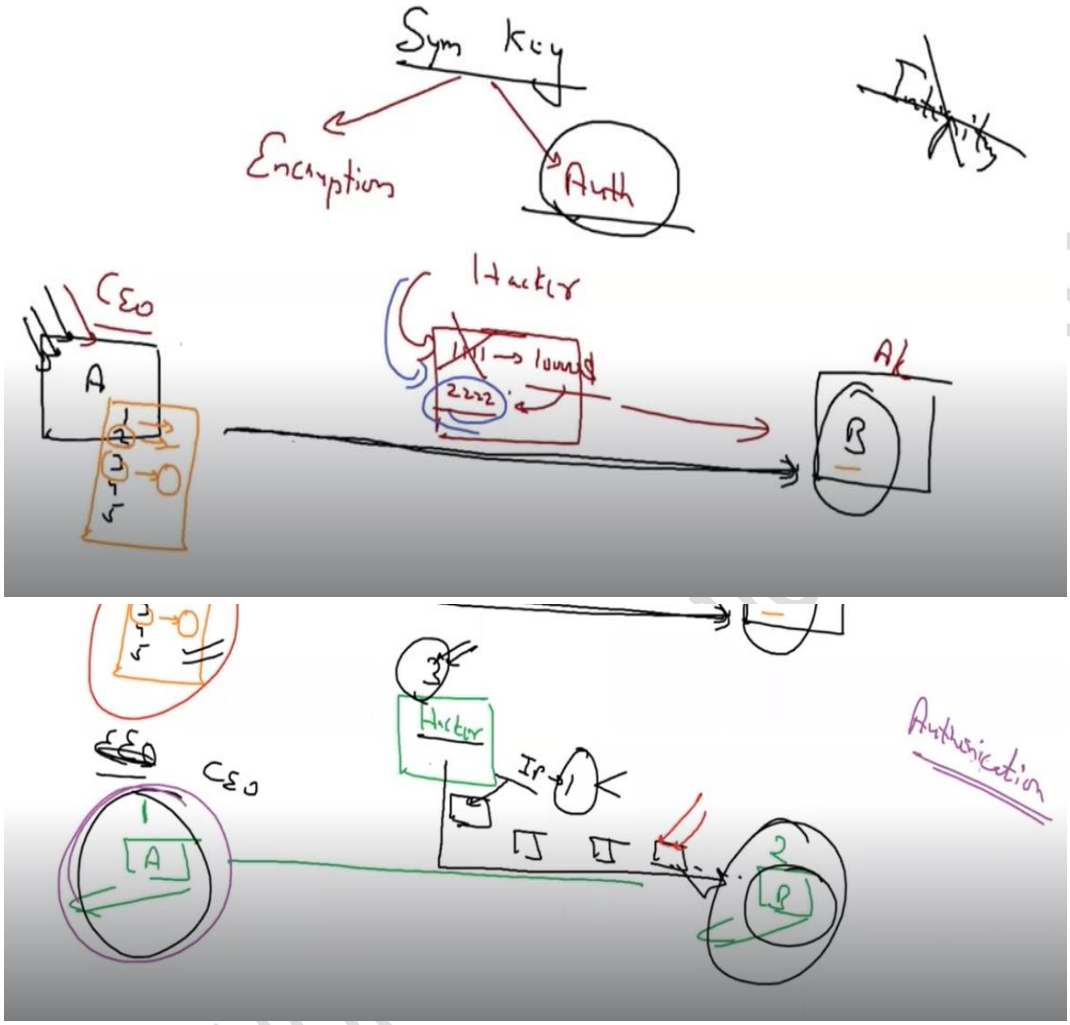
Summary 26-07-2022

- To Authenticate our users we have these Ways of cryptography
  - Key
    - Symmetric Key
    - Asymmetric Key
  - Hashing
  - MAC(Message Authentication Code)



- Till now, we have done encryption with the symmetric key, But we can also use a "symmetric key" for **authentication**.
  If 'A' have to transfer encrypted data to 'B', Then 'A' will first create a symmetric key and will encrypt his data with that key, Now 'A' will transfer this symmetric key to 'B' in some way(via n/w), Now both 'A' and 'B' have the symmetric encryption key, No one else other than them have this key, So Now If 'A' transfer an encrypted data to 'B', then only 'B' will be able to decrypt it, No one else can decrypt it, As they don't have the encryption key.
- Now let's say between 'A' and 'B' there are some hackers who have replaced A's data with their data and sent it to 'B', when 'B' tries to decrypt it, He will not be able to do so, because this (hacker data) is encrypted by some other key (hacker key),

- But if B can decrypt it then it means it was sent by “A” only, So like this “B” can authenticate “A”. And “B” can be sure that this data is sent by “A”.



- In **Symmetric key encryption**, We can encrypt and decrypt the data with the same key, But in **hashing**, we can convert “plain text” into its “hash value” but there is no way to get back the “plain text” from the “hash value”.

- In Symmetric key, we had the issue that if someone gets my “key” then he can get back the original plain text, Which is very critical for us, But in **hashing**, if they even get my hash value, they don’t have a way to get back the plain text from the “hash value”.