1. Before we have discussed about Encryption Algorithms (DES, DES3, AES, RSA) and Symmetric Key Encryption (DES, DES3, AES).
2. Encryption is usually used for making data unreadable for 3rd party.  
   The encrypted data when wired over network may be easily changed or compromised.   
   In that case, the receiver will not be able to read the original data or reading it with an error.

A diagram of a blockchain

Description automatically generated

1. **That is where Hash comes in**.
   1. Hash is fixed length String.
   2. Its length can be 128, 160-bit etc. based on the algorithm used to create it.
2. **How do Hash Algo works**?
   1. It takes **input data of any length** such assingle word, a line, paragraph or a file etc.
   2. When the hash function is applied, it produces **fixed length hash**.
   3. This hash is **irreversible** because from this hash we can not go back to the input data.
   4. Why?
   5. Because same hash may be created for different inputs.  
      That is why Hash Function is One-Way.
3. **Characteristics of Hash**?
   1. We cannot go back to the input.
   2. A slight change (a single character) in the input, the output hash will also change.
   3. The hash function does not require any key. Only Hash Function and Input enough.
      1. Still there are some hash functions which require keys.
4. **Why do we need Hash and Hash Functions?**
   1. Hash is used to check the integrity of the data transferred to the receiver by the receiver side.
   2. Original Data is hashed to get a hash value then that hash value and original data are sent to the receiver.
   3. The receiver will calculate the hash of the received original data using same hash function.
   4. If the received hash and recalculated one are same, it means data was not tampered.
   5. **NOTE**: If we want to secure the data, we can encrypt the data and encryption happens before generating hash value.
   6. As we discussed there are hash functions which use key to calculate hash value.
   7. **That is purpose of the key in hashing**?
   8. 
   9. In case of hash key, the sender will use the key to generate the hash and the receiver will also use the same key to recalculate the hash.   
      If received and recalculated hash both are same, it means two things now
      1. **Integrity Of data**: It means the data was not tampered as we discussed above.
      2. **Authentication of Sender**: As receiver generated the same hash using the same key otherwise it was not possible to generate same hash value so it means sender is the expected one.c
5. Jatin: Why Key in Hash Function?

### **Case 1 (Hash Without a Key)** If the sender generates a hash **without** using a secret key, an adversary (e.g., a man-in-the-middle) can intercept the encrypted data, replace it with their own altered version, and compute a new hash for that altered data. Because the hash calculation does not require a key, the receiver’s verification process will simply check the hash against the data and find a match—even though the data has been tampered with. Consequently, there is no assurance of data authenticity or protection against tampering in a keyless hashing scenario.

### **Case 2 (Hash With a Key)** When the sender uses a **secret key** to generate the hash (turning it into a Message Authentication Code, or MAC), the receiver must use the **same key** to verify the hash. If an adversary attempts to alter the data and recompute the hash, they will fail because they do not possess the key needed to produce a valid hash. As a result, the receiver will detect any mismatch and conclude the data was tampered with. This key-based approach provides both **data integrity** (the data has not been altered) and **authenticity** (the sender is verified).