Crypto Primitives for Blockchain

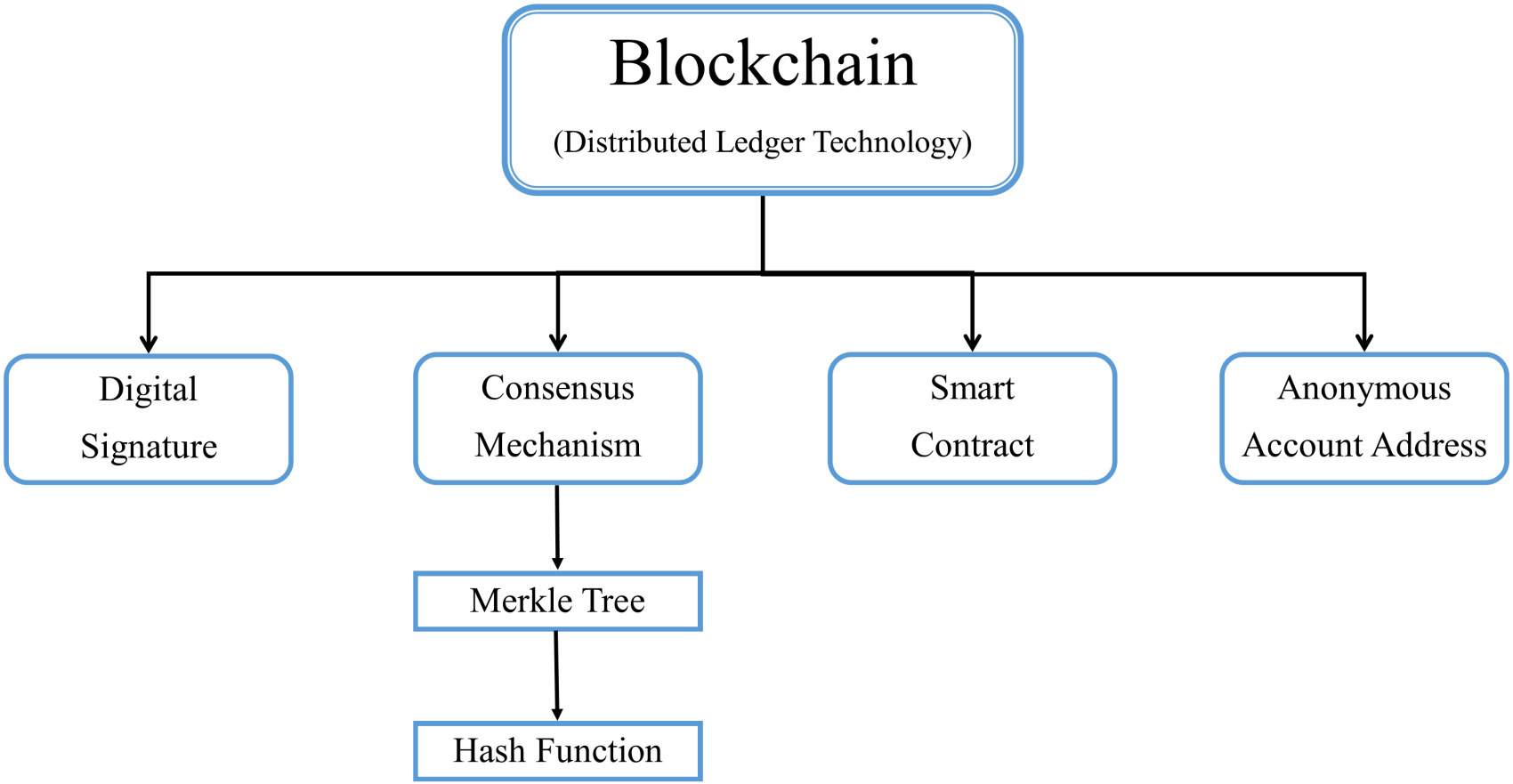
Bitcoin and Cryptocurrency Technologies
(CS662)

M.Tech. I, Semester II

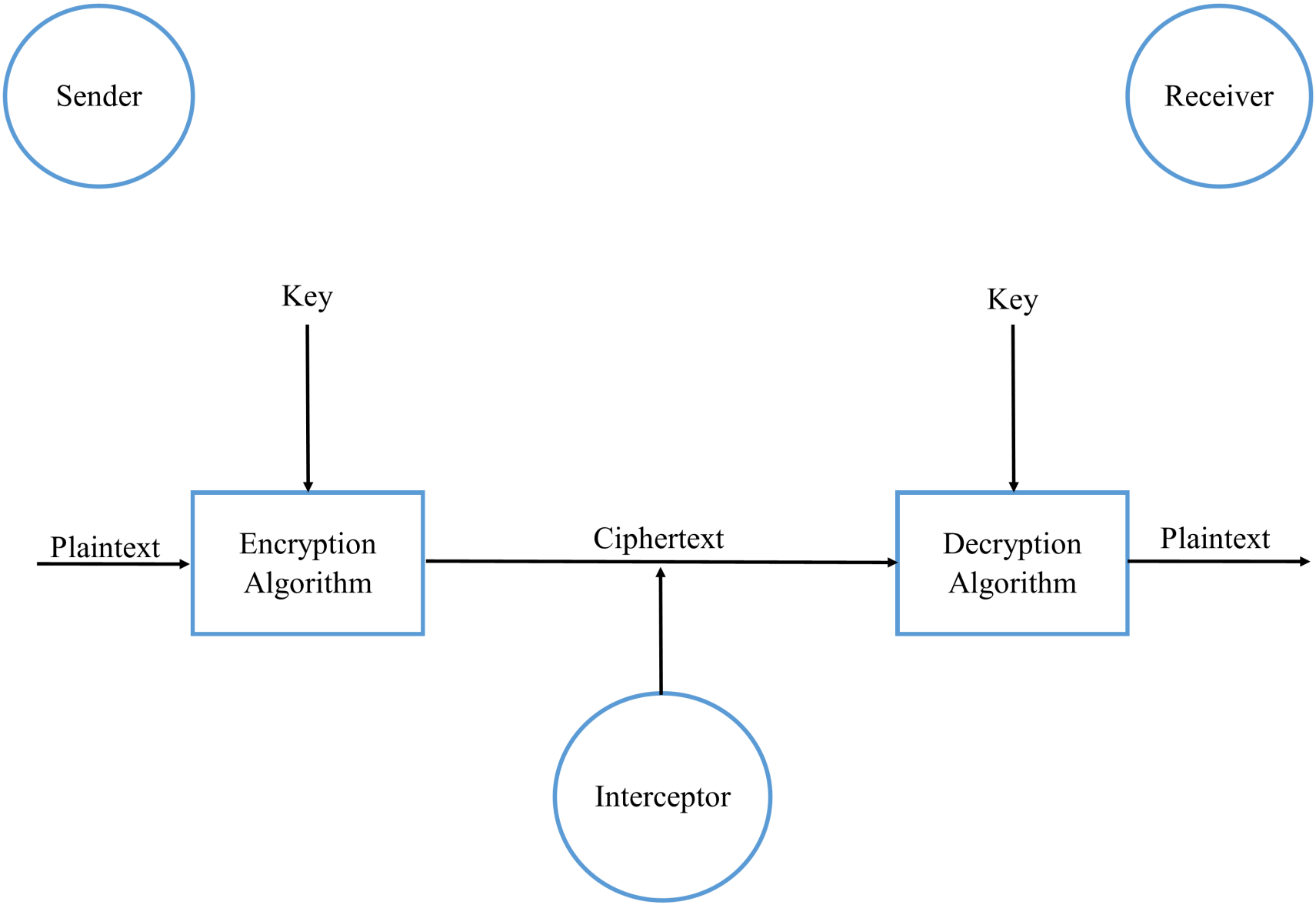
Department of Computer Science and Technology
S.V.National Institute of Technology-Surat

January 16, 2023

Hierarchy of Crypto Primitives for Blockchain



Symmetric Key Cryptography



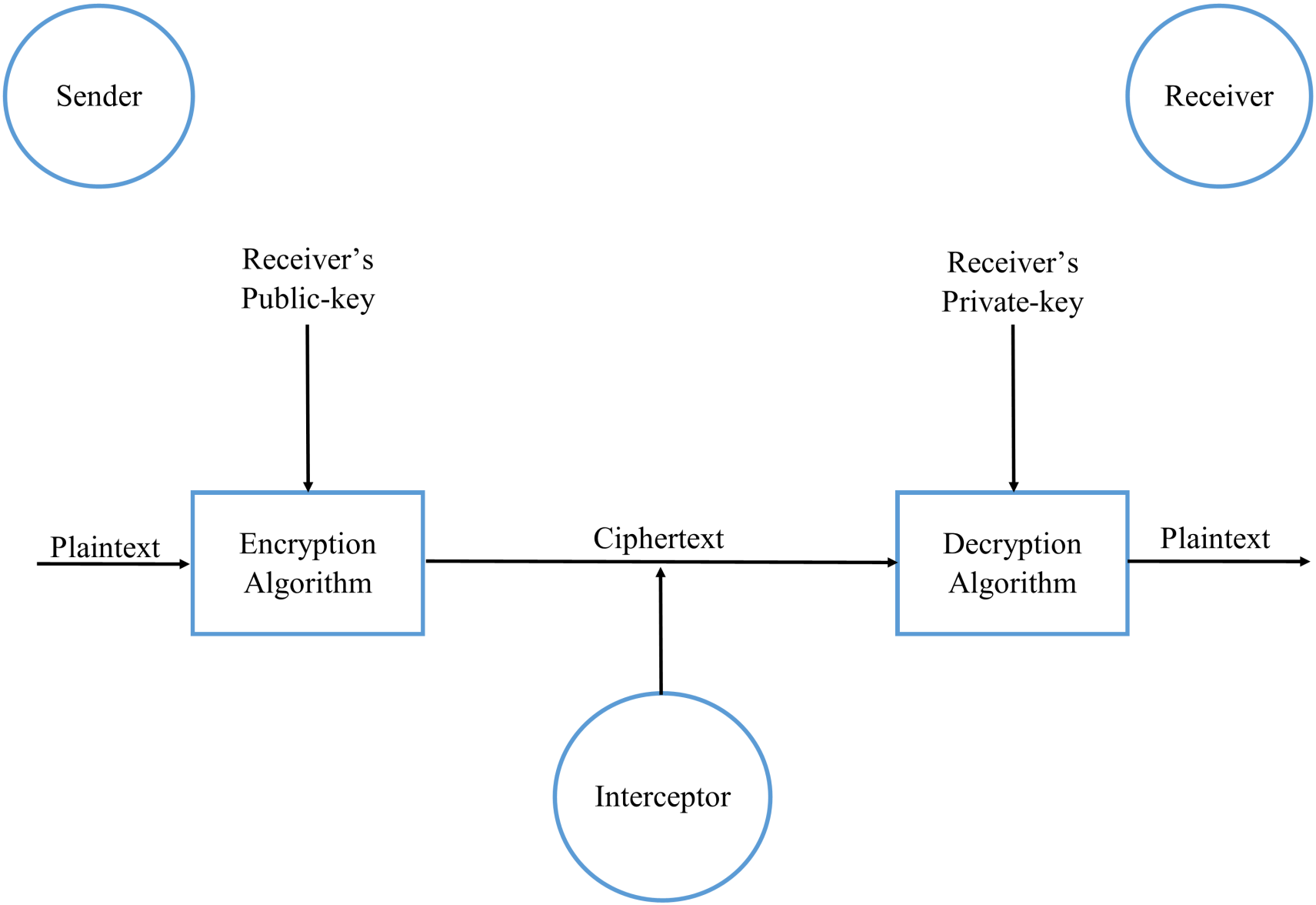
Advantages of Symmetric Key Cryptography

- Symmetric key encryption can be highly secure when it employs a secure algorithm.
- It's pretty simple to encrypt and decrypt symmetric key data, resulting in excellent reading and writing performance.
- Because of their security and speed benefits, symmetric encryption algorithms like AES have become the gold standard of data encryption.
- Requires low computer resources.

Disadvantages of Symmetric Key Cryptography

- The most significant drawback of symmetric key encryption is that the key must be communicated to the party with which you share data.
- When someone obtains a symmetric key, they can decode anything that has been encrypted with that key. When two-way communications are encrypted by symmetric encryption, both sides of the conversation are vulnerable.
- The message's origin and authenticity cannot be guaranteed.

Public Key Cryptography



Advantages of Public Key Cryptography

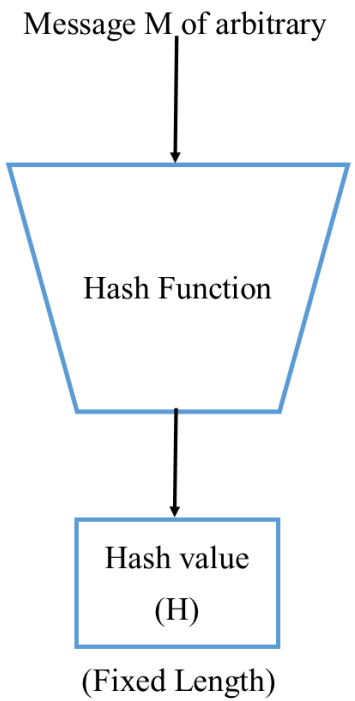
- It allows message authentication.
- Solves the problem of distributing keys for encryption, with everyone publishing their public keys, while private keys being kept secret.
- Allows for non-repudiation.

Disadvantages of Public Key Cryptography

- Comparatively slow process.
- Public Key Encryption also is weak towards man in the middle attack. In this attack a third party can disrupt the public key communication and then modify the public keys.
- Basically, no one absolutely knows that a public key belongs to the individual it specifies, which means that users will have to verify that their public keys truly belong to them.
- When you lose your private key, your received messages will not be decrypted.
- If your private key is identified by an attacker, all of your messages can be read by him/her.

Hash Function

- A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length.



Features of Hash Functions

- Fixed Length Output (Hash Value)
  - Hash function converts data of arbitrary length to a fixed length. This process is often referred to as hashing the data.
  - In general, the hash is much smaller than the input data, hence hash functions are sometimes called compression functions.
  - Since a hash is a smaller representation of a larger data, it is also referred to as a digest.
  - Hash function with n bit output is referred to as an n-bit hash function. Popular hash functions generate values between 160 and 512 bits.
- Efficiency of Operation
  - Generally for any hash function h with input x, computation of h(x) is a fast operation.
  - Computationally hash functions are much faster than a symmetric encryption.

Properties of Hash Functions

- Pre-Image Resistance
  - It should be computationally hard to reverse a hash function.
  - This property protects against an attacker who only has a hash value and is trying to find the input.
- Second Pre-Image Resistance
  - Given an input and its hash, it should be hard to find a different input with the same hash.
  - This property of hash function protects against an attacker who has an input value and its hash, and wants to substitute different value as legitimate value in place of original input value.
- Collision Resistance
  - It should be hard to find two different inputs of any length that result in the same hash. This property is also referred to as collision free hash function.
  - This property makes it very difficult for an attacker to find two input values with the same hash.
  - If a hash function is collision-resistant then it is second pre-image resistant.

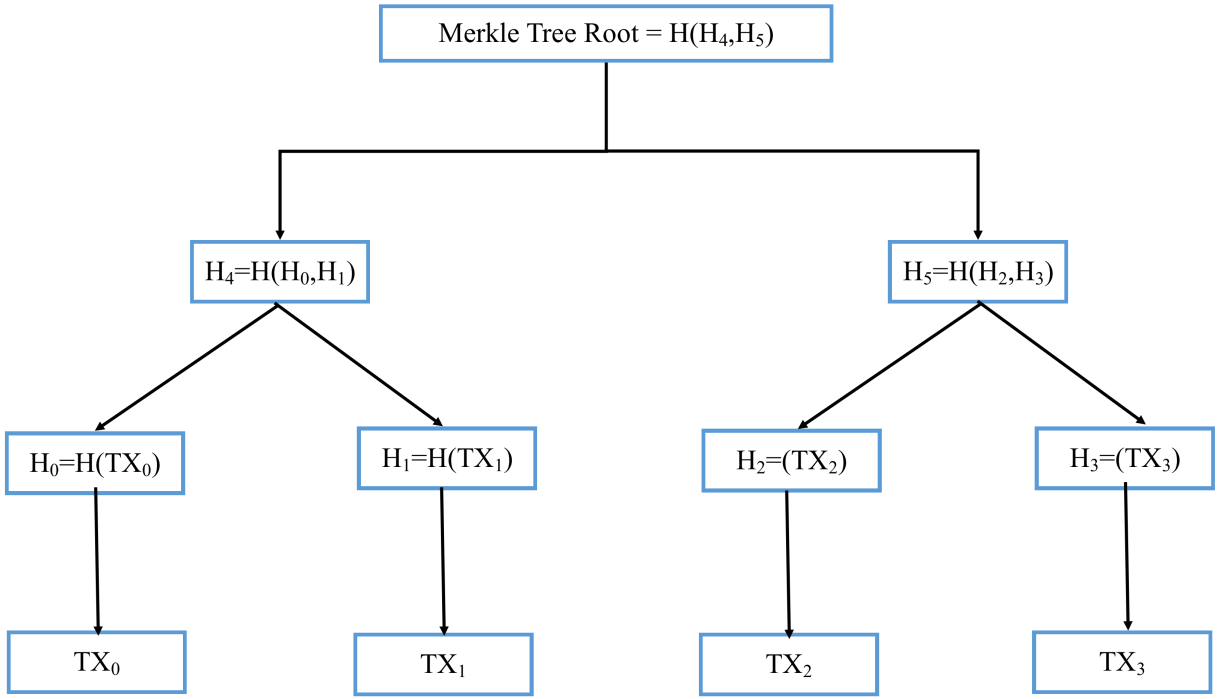
Use of Hash Functions

- Password Storage
- Data Integrity Check
- Proof of Work in Blockchain

Well Known Hash Functions

- MD5
  - output 128 bits
  - collision resistance completely broken by researchers in China in 2004
- SHA1
  - output 160 bits
  - no collision found yet, but method exist to find collisions in less than 2^{80}
  - considered insecure for collision resistance
  - one-wayness still holds
- SHA2 (SHA-224, SHA-256, SHA-384, SHA-512)
  - outputs 224, 256, 384, and 512 bits, respectively
  - No real security concerns yet

Merkle tree



Merkle tree summarizing all transactions in a block to produce a digital fingerprint of the entire set of transactions, thereby enabling a user to verify whether a transaction is included in a block.