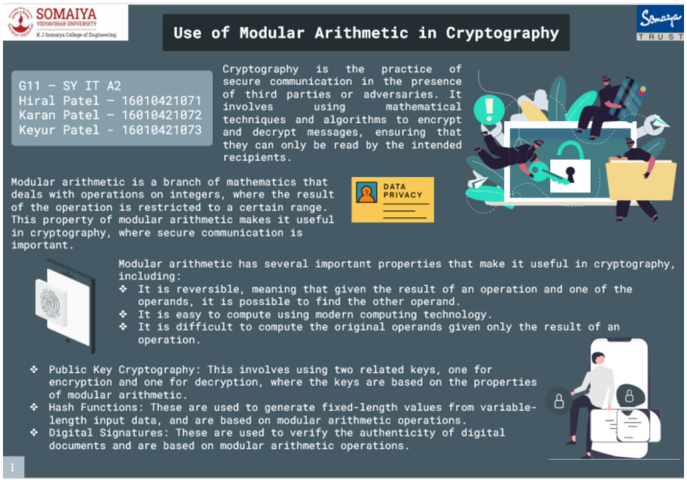
<u>Use of Modular Arithmetic in Cryptography</u>

Information Theory and Coding
IA2 – SY IT A2
Hiral Patel – 16010421071
Karan Patel – 16010421072
Keyur Patel – 16010421073

Presentation Images:

Presentation Explanation for page 1:

<u>Images:</u>



<u>Content:</u>

What is Modulo Arithmetic?

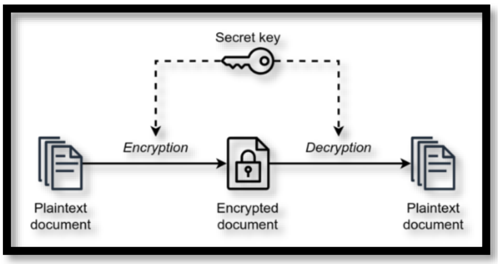
Modular arithmetic is a type of arithmetic in which the operands are numbers that are modulo a certain number. In other words, the operands are reduced to a common denominator.

Why is Modular Arithmetic used in Cryptography?

- The modulo operation is used in cryptography to create secure communications.
- It is used to create a shared secret between two parties that can be used to encrypt and decrypt data.
- It is difficult to solve modular arithmetic problems, but easy to verify the results. This makes it a good choice for cryptographic algorithms.
- Modular arithmetic is a fundamental concept in cryptography that is used extensively in many cryptographic algorithms.

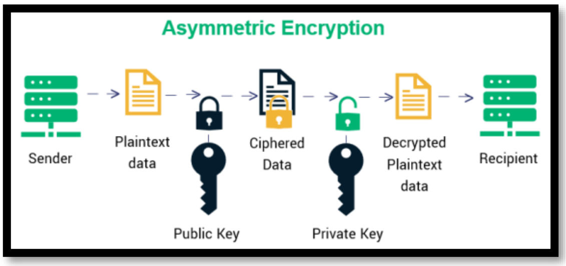
Types of Cryptography:

1) **<u>Symmetric Key Cryptography:</u>**



- Symmetric cryptography, also known as secret-key cryptography, uses a single secret key to encrypt and decrypt messages.
- The sender and the receiver both use the same key to encrypt and decrypt messages, making it faster and more efficient than asymmetric cryptography.

2) **<u>Asymmetric Key Cryptography:</u>**



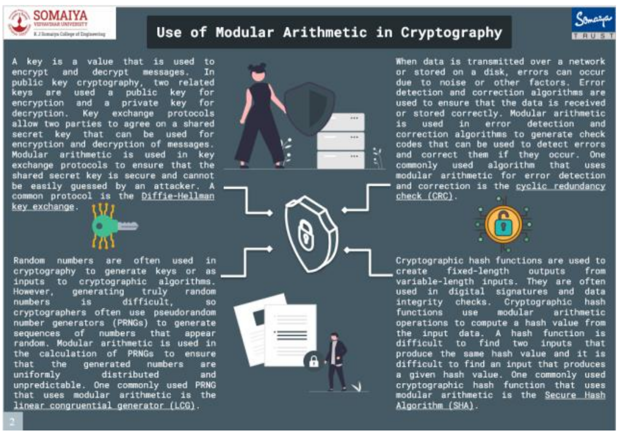
- Asymmetric cryptography, also known as public-key cryptography, uses a pair of keys: a public key and a private key.
- The public key is used to encrypt messages, and the private key is used to decrypt them.

3) **<u>Hash Function Cryptography:</u>**

- There is also a third type of cryptography known as hash functions, which are used to generate fixed-length outputs from arbitrary-length inputs.
- Hash functions are commonly used in digital signatures and to ensure the integrity of data. However, they are not used for encryption or decryption of messages.

Presentation Explanation for page 2:

<u>Images:</u>



Content:

Uses of Modular Arithmetic in Cryptography:

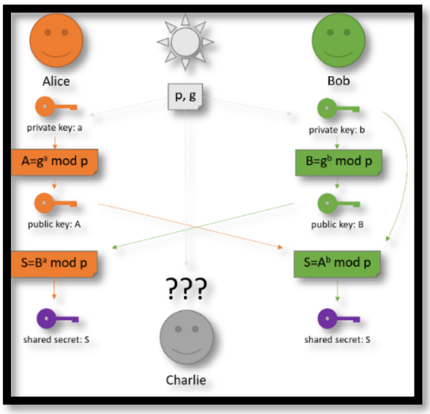
1) **Public key Cryptography:-**

- When someone wants to send a message to the recipient, they use the recipient's public key to encrypt the message.
- Once the message is encrypted, only the recipient's private key can be used to decrypt it.
- This ensures that only the intended recipient can read the message, even if the message is intercepted by an attacker.

One such Algorithm used in Public key Cryptography is Diffie-hellman key exchange.

Diffie-hellman key Exchange

- The Diffie-Hellman key exchange (also known as exponential key exchange) is a method for securely exchanging cryptographic keys over an insecure channel.
- It works by allowing two parties (Alice and Bob) to agree on a shared secret key without any other party being able to intercept the key or learn anything about it.
- The key exchange involves the following steps −
  - Alice and Bob agree on two large prime numbers, **p** and **g**, and a public key exchange algorithm.
  - Alice chooses a secret integer, **a**, and computes $\mathbf{A = g^a \bmod p}$. She sends **A** to Bob.
  - Bob chooses a secret integer, **b**, and computes $\mathbf{B = g^b \bmod p}$. He sends **B** to Alice.
  - Alice computes $\mathbf{S = B^a \bmod p}$. Bob computes $\mathbf{S = A^b \bmod p}$.
  - Alice and Bob now both have the shared secret key **S**, which they can use to establish a secure communication channel.



2) **Hash Function Cryptography:-**

- A hash function is a mathematical function that takes in an input (message or data) and outputs a fixed-size string of characters, known as a hash value or message digest.

- Division Modulo Method is the simplest method of hashing.
- In this method, we divide the element with the size of the hash table and use the remainder as the index of the element in the hash table.

Applications of Hash function in Cryptography:

- Hash functions are used in a variety of applications, such as password storage, digital signatures, and message authentication codes (MACs).
- In password storage, a hash function is used to transform a user's password into a fixed-length hash value, which is stored in a database.
- When the user logs in, the system hashes the entered password and compares it to the stored hash value to determine if the password is correct.
- In digital signatures, a hash function is used to generate a message digest of a document, which is then encrypted using the sender's private key. The recipient can use the sender's public key to decrypt the message digest and verify the authenticity of the document.

3) **Linear Congruential Generator:-**

- A linear congruential generator (LCG) is a simple algorithm used to generate a sequence of pseudo-random numbers.
- It is not typically used for key exchange as it is considered insecure due to predictable patterns in the generated numbers.
- However, the security of this key exchange method is compromised by the fact that LCGs can exhibit predictable patterns in their output.
- An attacker who can predict the output of the LCG could determine the shared secret key and intercept messages.

4) **Cyclic Redundancy Check(CRCs) :-**

Cyclic Redundancy Check (CRC) is an error detection technique used in digital communication. It is used to verify the integrity of data transmitted over a network or stored in memory.

Here is an example of how CRC works:

Suppose we want to transmit the message "**110101**" over a communication channel. To detect any errors in transmission, we can use CRC.

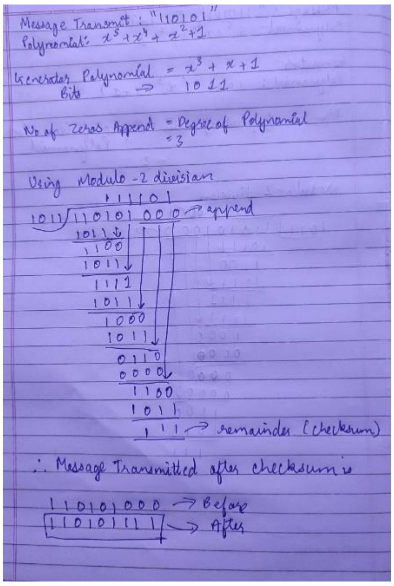
The CRC algorithm uses a divisor polynomial to generate a checksum, which is appended to the message.

1. Choose a generator polynomial. The generator polynomial is a fixed value that is agreed upon by both the sender and receiver. It is usually a binary number represented as a polynomial. For example, the generator polynomial could be **1011**, which represents $\mathbf{x^3 + x + 1}$**.**

2. Append zeros to the message. The number of zeros to be appended is equal to the degree of the generator polynomial. In our example, we need to append three zeros to the message "**110101**" to get "**110101000**".

3. Divide the message by the generator polynomial using modulo-2 division. This generates a remainder that is used as the checksum. In our example, the message "**110101000**" divided by the generator polynomial "**1011**" gives a remainder of "**111**".

4. Append the checksum to the original message. In our example, the message with the checksum is "**110101111**".



<u>References:</u>

1) https://www.geeksforgeeks.org/linear-congruence-method-for-generating-pseudo-random-numbers/

2) https://www.tutorialspoint.com/cryptography/cryptography_digital_signatures.htm

3) https://technicalsand.com/hashing-in-data-structure/

4) https://www.techtarget.com/searchsecurity/definition/Diffie-Hellman-key-exchange

5) https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

6) https://www.educba.com/diffie-hellman-key-exchange-algorithm/

7) https://www.tutorialspoint.com/what-is-modular-arithmetic-in-information-security#:~:text=Modular%20arithmetic%20enables%20us%20to,different%20groups%20which%20can%20work.