Encryption Systems: RSA, Paillier, and ElGamal

Your Name

June 23, 2024

1 Introduction

This report covers the implementation and analysis of three encryption systems: RSA, Paillier, and ElGamal. Each section includes a description of the algorithm, the key generation process, the working mechanism, security features, and time complexity analysis.

2 RSA Encryption

2.1 Algorithm Description

RSA is a public-key encryption algorithm that relies on the mathematical properties of prime numbers and modular arithmetic. It involves generating a pair of keys: a public key for encryption and a private key for decryption.

2.2 Key Generation Process

1. Select two distinct prime numbers p and q.
2. Compute $n = p \times q$.
3. Compute the totient function $\phi(n) = (p - 1) \times (q - 1)$.
4. Choose an integer e such that $1 < e < \phi(n)$ and $\gcd(e, \phi(n)) = 1$.
5. Compute d as the modular multiplicative inverse of e modulo $\phi(n)$.

2.3 Working Mechanism

- **Encryption:** $c = m^e \mod n$
- **Decryption:** $m = c^d \mod n$

2.4 Security Features

RSA's security is based on the difficulty of factoring large composite numbers.

2.5 Time Complexity

The time complexity for key generation is $O((\log n)^3)$, and for encryption/decryption, it is $O((\log n)^2)$.

3 Paillier Encryption

3.1 Algorithm Description

Paillier encryption is a probabilistic asymmetric algorithm for public-key cryptography. It is based on the decisional composite residuosity assumption.

3.2 Key Generation Process

1. Select two large prime numbers p and q.
2. Compute $n = p \times q$ and n^2.
3. Select $g \in Z^*_{n^2}$ such that $gcd(L(g^\lambda \mod n^2), n) = 1$, where $\lambda = \text{lcm}(p - 1, q - 1)$.
4. Compute $\mu = (L(g^\lambda \mod n^2))^{-1} \mod n$, where $L(u) = \frac{u-1}{n}$.

3.3 Working Mechanism

- **Encryption:** $c = g^m \cdot r^n \mod n^2$, where r is a random integer in Z^*_n.
- **Decryption:** $m = \frac{L(c^\lambda \mod n^2) \cdot \mu \mod n}{}$.

3.4 Security Features

The security of Paillier encryption relies on the difficulty of distinguishing composite residuosity classes.

3.5 Time Complexity

The time complexity for key generation, encryption, and decryption is $O((\log n)^2)$.

4 ElGamal Encryption

4.1 Algorithm Description

ElGamal encryption is an asymmetric key encryption algorithm for public-key cryptography based on the Diffie-Hellman key exchange.

4.2 Key Generation Process

1. Choose a large prime p and a generator g of the multiplicative group of integers modulo p.
2. Select a random integer x such that $1 < x < p - 1$.
3. Compute $y = g^x \mod p$.

4.3 Working Mechanism

- **Encryption:** $c_1 = g^k \mod p$, $c_2 = m \cdot y^k \mod p$, where k is a random integer.
- **Decryption:** $m = c_2 \cdot (c_1^x)^{-1} \mod p$.

4.4 Security Features

The security of ElGamal encryption is based on the difficulty of computing discrete logarithms in a finite field.

4.5 Time Complexity

The time complexity for key generation, encryption, and decryption is $O((\log p)^3)$.

5 Conclusion

This report presented the implementation details, working mechanisms, and security features of RSA, Paillier, and ElGamal encryption systems. Each system has its own advantages and use cases in the field of cryptography.