Paillier Encryption

Based on:

Other References:

Encryption Scheme
Paillier Encryption is an asymmetric encryption scheme $< K, E, D >$

Algorithm $K()$
choose large primes $p, q$
$n \leftarrow pq$
g $\leftarrow \mathbb{Z}_n^*$
$pk \leftarrow (n, g)$
sk $\leftarrow (p, q)$
return $(pk, sk)$

Note: $g$ must satisfy 2 properties, to be discussed later

Algorithm $E(m)$
if $m \geq n$, return ⊥
r $\leftarrow \mathbb{Z}_n^*$ / {0}
c $\leftarrow g^m * r^n \mod n^2$
return $c$

Algorithm $D(c)$
if $c \geq n^2$, return ⊥
$\lambda \leftarrow \text{lcm}(p - 1)(q - 1)$
m $\leftarrow L(c^\lambda \mod n^2) \mod n$
return $c$

Define $L(u) = (u - 1)/n$
How Does It Work?

If \( g \in \mathbb{Z}_{n^2} \), Define \( \epsilon_g \):
\[
\mathbb{Z}_n \times \mathbb{Z}_n^* \rightarrow \mathbb{Z}_n^*
\]
\[(x, y) \rightarrow g^x y^n \bmod n^2
\]

- Notice: \( |\mathbb{Z}_n^*| = \phi(n^2) = n \cdot \phi(n) = |\mathbb{Z}_n \times \mathbb{Z}_n^*|
- For any \( w \in \mathbb{Z}_n^* \), \( w^\lambda \equiv 1 \bmod n \) and \( w^{n\lambda} \equiv 1 \bmod n^2
- Remember: \( \phi(n) = (p - 1)(q - 1) \)

Algorithm \( E(m) \)

if \( m \geq n \), return \( \perp \)
\( r \leftarrow \mathbb{Z}_n^*/\{0\} \)
\( c \leftarrow g^m \cdot r^n \bmod n^2 \)
return \( c \)

How Does It Work?

- Decryption:
  - Since \( g \in \mathbb{Z}_n^* \), \( g^\lambda \bmod n^2 \) yields an element of \( \mathbb{Z}_{n^2} \) congruent to \( 1 \bmod n \)
  - \( L(g^\lambda \bmod n^2) \) calculation is feasible; this value will be \( \in \mathbb{Z}_n^* \)
  - Need \( \mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n \)
- \( c \) is also \( \in \mathbb{Z}_n^* \)
- \( m \equiv L(c^\lambda \bmod n^2) \cdot \mu \bmod n \)

Algorithm \( D(c) \)

if \( c \geq n^2 \), return \( \perp \)
\( \lambda \leftarrow \text{lcm}(p - 1)(q - 1) \)
\( m \leftarrow \frac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} \bmod n \)
return \( c \)

Define \( L(u) = (u - 1)/n \)
Specifics of Key Generation: Choosing \( g \)

1. The order of \( g \) needs to be a nonzero multiple of \( n \in \mathbb{Z}_{n^2}^* \) This makes \( \epsilon_g \) bijective.
   - Order of a group element is defined as the smallest positive integer \( m \) such that \( g^m = e \), \( e \) being the identity element of the group.

2. \( L(g^\lambda \mod n^2) \) should not be \( \equiv \to \) to a multiple of \( p \) or \( q \mod n \)
   - This is because there must exist modular multiplicative inverse \( \mu \in \mathbb{Z}_n^* \) s.t.
     \[
     \mu = (L(g^\lambda \mod n^2))^{-1} \mod n
     \]

Decisional Composite Residuosity Assumption

A number \( z \) is a \( n \)-th residue modulo \( n^2 \) if:

\[
\exists \ a \ number \ y \in \mathbb{Z}_{n^2}^* \ such \ that: \quad z = y^n \mod n^2
\]

- DCRA essentially states that (if \( n \) and \( z \) are known) the problem of “deciding \( n \)-th residuosity”, or distinguishing \( n \)-th residues from non \( n \)-th residues, is computationally hard.

- In other words, finding whether \( \exists \ y \) to satisfy the above equation is hard.
Computational Composite Residuosity Assumption

• CCRA states that the n-th Residuosity Class Problem in base g is computationally hard.

• This problem is defined as the problem of computing the class function in base g:

  For a given \( w \in \mathbb{Z}_n^* \), find the integer \( x \in \mathbb{Z}_n \) for which

  \[ \exists y \in \mathbb{Z}_n^* \text{ such that } \epsilon_g(x, y) = w \]

• This is exactly what we’re doing in decrypting a given ciphertext.

• If DCRA is true, CCRA is true

Security

• DCRA implies IND-CPA security

• CCRA implies one-wayness

• Later, a modified version of the scheme was proven secure against adaptive chosen-ciphertext attacks (IND-CCA2) in the random oracle model

• The new scheme was mathematically very similar

• m and r can be of more variable lengths

• The M used in encryption is basically the original m appended with a string that reflects the hashing of m and r using two random oracles.
Fun Properties

\( \forall m_1, m_1 \in \mathbb{Z}_n \) ...

- **Homomorphic Addition**
  \[ D(E(m_1)E(m_2) \mod n^2) = m_1 + m_2 \mod n \]

- In Fact:
  \[ D(E(m_1)g^{m_2} \mod n^2) = m_1 + m_2 \mod n \]

- **Self-blinding** follows:
  \[ D(E(m_1)g^{nr} \mod n^2) = m_1 + nr \mod n = m_1 \mod n \]

Homomorphic Addition

\[ D(E(m_1)E(m_2) \mod n^2) = m_1 + m_2 \mod n \]

\[
E(m_1) \equiv c_1 \equiv g^{m_1} \cdot r_1^n \mod n^2 \\
E(m_2) \equiv c_2 \equiv g^{m_2} \cdot r_2^n \mod n^2 \\
c_1 \cdot c_2 \equiv g^{m_1} \cdot g^{m_2} \cdot r_1^n \cdot r_2^n \equiv g^{m_1+m_2} \cdot (r_1r_2)^n \mod n^2
\]

- Since \( r_1, r_2 \in \mathbb{Z}_n^* \), then \( r_1 \cdot r_2 \in \mathbb{Z}_n^* \)
- This is the encryption of a new message, \( m_1 + m_2 \)
Fun Properties

∀m₁, m₁ ∈ ℤn ...

- **Homomorphic Multiplication**

  \[ D(E(m₁)^{m₂} \mod n^2) = D(E(m₂)^{m₁} \mod n^2) = m₁m₂ \mod n \]

- Also follows for nonnegative integer constants