Number Theory, Public Key Cryptography, RSA
Ahmet Burak Can
Hacettepe University
abc@hacettepe.edu.tr

The Euler Phi Function

- Definition:
  For a positive integer \( n \), if \( 0 < a < n \) and \( \gcd(a, n) = 1 \), \( a \) is relatively prime to \( n \).

- Definition:
  Given an integer \( n \), \( \varphi(n) \) is the number of positive integers less than or equal to \( n \) and relatively prime to \( n \).

The Euler Phi Function (cont.)

- Theorem: Formula for \( \varphi(n) \)
  - Let \( p \) be prime, \( e, m, n \) be positive integers
    1) \( \varphi(p) = p - 1 \)
    2) if \( \gcd(m, n) = 1 \), then \( \varphi(mn) = \varphi(m) \varphi(n) \)
    3) \( \varphi(p^e) = p^e - p^{e-1} \)
    4) If \( n = p_1^{e_1} p_2^{e_2} \ldots p_k^{e_k} \) then
       \[ \varphi(n) = n \left(1 - \frac{1}{p_1}\right) \left(1 - \frac{1}{p_2}\right) \ldots \left(1 - \frac{1}{p_k}\right) \]

Fermat’s Little Theorem

- Fermat’s Little Theorem
  - If \( p \) is a prime number and \( a \) is a natural number that is not a multiple of \( p \), then
    \[ a^{p-1} = 1 \pmod{p} \]
Euler's Theorem

- Euler's Theorem
  Given integer n>1, such that gcd(a,n)=1 then
  \[ a^{\varphi(n)} \equiv 1 \pmod{n} \]
- Corollary
  Given integer n>1, such that gcd(a,n)=1 then
  \[ a^{\varphi(n)-1} \pmod{n} \]
  is a multiplicative inverse of a mod n.

Consequence of Euler's Theorem

- Principle of Modular Exponentiation
  Given integer n>1, x, y, and a positive integers with gcd(a,n)=1. If x\equiv y (mod \varphi(n)), then
  \[ a^x = a^y (mod \ n) \]
- Proof idea:
  \[ a^x \equiv a^{(x+y) \varphi(n)} \equiv a^y (a^{\varphi(n)})^k \]
  by applying Euler's theorem we obtain
  \[ a^x = a^y (mod \ n) \]

Diffie-Hellman Key Exchange

- Diffie-Hellman proposed a cryptographic protocol to exchange keys among two parties in 1976.
  - Public parameters:
    - p: A large prime
    - g: Base (generator)
  - Secret parameters:
    - \( \alpha, \beta \in \{0, 1, 2, \ldots, p-2\} \)
  - Alice computes \( g^\alpha \pmod{p} \)
  - Bob computes \( g^\beta \pmod{p} \)
  - \( g^{\alpha\beta} \pmod{p} \)
  - Alice computes \( g^\beta \pmod{p} \)
  - Bob computes \( g^\alpha \pmod{p} \)
  - Key \( K = g^{\alpha\beta} \pmod{p} \)

Security of Diffie-Hellman

- Discrete Logarithm Problem (DLP):
  - Given p, g, \( g^x \pmod{p} \), what is x?
  - easy in Z, hard in \( Z_p \)
- Diffie-Hellman Problem (DHP):
  - Given p, g, \( g^x \pmod{p} \), \( g^y \pmod{p} \), what is \( g^{\alpha\beta} \pmod{p} \)?
  - DHP is as hard as DLP.
Commutative Encryption

- Definition:
  An encryption scheme is commutative if
  \[ E_{K_1}(E_{K_2}[M]) = E_{K_2}(E_{K_1}[M]) \]

  Given a commutative encryption scheme, then
  \[ D_{K_1}(D_{K_2}[E_{K_1}[M]]) = M \]

- Most symmetric encryption scheme are not commutative such as DES and AES.

Asymmetric Encryption Functions

- An asymmetric encryption function:
  - Encryption (K) and decryption (K^-1) keys are different.
  - Knowledge of the encryption key is not sufficient for deriving the decryption key efficiently.
  - Hence, the encryption key can be made “public”.

Pohlig-Hellman Exponentiation Cipher

- A commutative exponentiation cipher
  - encryption key (e, p), where p is a prime
  - decryption key (d, p), where ed ≡ 1 (mod (p-1)) or in other words d = e^-1 (mod (p-1))
  - to encrypt M, compute \( C = M^e \mod p \)
  - to decrypt C, compute \( M = C^d \mod p = M^{ed} \mod p \)

Public Key Encryption

- Each party has a PAIR (K, K^-1) of keys:
  - K is the public key
  - K^-1 is the private key

\[ \text{D}_K^{-1}[E_K[M]] = M \]

- The public-key K may be made publicly available.
- Many can encrypt with the public key, only one can decrypt.
- Knowing the public-key and the cipher, it is computationally infeasible to compute the private key.
Solutions with Public Key Cryptography

- Key distribution solution:
  - Alice makes her encryption key K public
  - Everyone can send her an encrypted message:
    \[ C = E_k(M) \]
  - Only Alice can decrypt it with the private key K⁻¹:
    \[ M = D_k^{-1}(C) \]

- Source Authentication Solution:
  - Only Alice can “sign” a message, using K⁻¹.
  - Anyone can verify the signature, using K.
  - Only if such a function could be found...

RSA Algorithm

- Invented in 1978 by Ron Rivest, Adi Shamir and Leonard Adleman
  - Security relies on the difficulty of factoring large composite numbers
  - Essentially the same algorithm was discovered in 1973 by Clifford Cocks, who works for the British intelligence

RSA Public Key Crypto System

- Choose large primes p, q
  - Compute \( n = pq \) and \( \varphi(n) = (q-1)(p-1) \)
- Choose e, such that \( \gcd(e, \varphi(n)) = 1 \).
  - Take e to be a prime
  - Compute \( d = e^{-1} \mod \varphi(n) \) ed = 1 \( \mod \varphi(n) \)
  - Public key: n, e
  - Private key: d
- Encryption: \( C = E(M) = M^e \mod n \)
  - Decryption: \( D(C) = C^d \mod n = M \)

RSA Encryption

- Encryption: \( C = E(M) = M^e \mod n \)
- Decryption: \( D(C) = C^d \mod n \)
- Why does it work?
  \[ D(M) = (M^e)^d \mod n = M^{ed} \mod n \]
  \[ = M^{k\varphi(n)+1} \mod n, \quad (for \ some \ k) \]
  \[ = (M^{\varphi(n)})^k \cdot M \mod n \]
  \[ = M \]

- RSA problem: Given n, e, \( M^e \mod n \), what is M?
  - Computing d is equivalent to factoring n.
  - The security is based on difficulty of factoring large integers.
Let $e = 37$, then
- $n = 77, \varphi(n) = 60$
- $d = 13$ (since $ed \mod 60 = 1$)

Let $M = 15$, then $C \equiv M^e \mod n$

- $C \equiv 15^{37} \mod 77 = 71$
- $M \equiv C^d \mod n$

$M \equiv 71^{13} \mod 77 = 15$

**RSA Implementation**

- The security of RSA depends on how large $n$ is, which is often measured in the number of bits for $n$.
  - Current recommendation is 1024 bits for $n$.
  - $p$ and $q$ should have the same bit length, so for 1024 bits RSA, $p$ and $q$ should be about 512 bits.
  - $p-q$ should not be small.
    - In general, $p,q$ randomly selected and then tested for primality
    - Many implementations use the Rabin-Miller test (probabilistic test)