CSE 311 Foundations of Computing I

Lecture XX

RSA Encryption and Decryption

Autumn 2011

Public Key Encryption/Decryption

- Bob wants people to be able to send him secret messages so he creates
  - An encryption key PK which he makes public along with an encryption algorithm E_{PK}
  - A decryption key SK which he keeps secret along with a decryption algorithm D_{SK}
  - Requirement: $D_{SK}(E_{PK}(m))=m$
- If Alice wants to send a message m to Bob she
  - Computes $C=E_{PK}(m)$ and sends it to Bob
  - Bob computes $D_{SK}(C)$ which equals m

Public Key Encryption/Decryption

- First developed in 1976 by Diffie-Hellman using number theory problems
- Rivest-Shamir-Adleman (RSA) came up with a simpler number theory-based method that is in use today

RSA Encryption/Decryption

- Bob choose two random large prime numbers p and q and computes N=pq
  - Usually p and q are 512 or 1024 bits long
  - We won't worry about how Bob does this
- Bob chooses a some big odd number e < N
- PK=(e,N): $E_{PK}(m)=m^e \bmod N$
  - Computable quickly using fast modular exponentiation
- SK=(d,p,q): $D_{SK}(C)=C^d \bmod N$ where d depends on e, p, and q
  - Computable quickly using fast modular exponentiation
- We need that $(m^e)^d=m^{ed} \equiv m \pmod N$
  - How does Bob find such a d?

Modular Equations and $\mathbb{Z}^{\times}_N$

Recall: If gcd(a,N)=1 then we can solve $ax\equiv b \pmod N$ for a unique value x between 0 and N-1

Idea: Apply Euclid's algorithm for gcd(a,N) and then substitute back to write 1=sa+tN where 0<s<N
Then $sa \equiv 1 \pmod N$ so $x \equiv sax \equiv sb \pmod N$

Definition: Let $\mathbb{Z}^{\times}_N = \{ a : 0<a<N \text{ and } \gcd(a,N)=1\}$

We will show that properties of $\mathbb{Z}^{\times}_N$ will help us understand exponentiation modulo N

Properties of $\mathbb{Z}^{\times}_N = \{ a : 0<a<N \text{ and } \gcd(a,N)=1\}$

- (Multiply) If $a,b \in \mathbb{Z}^{\times}_N$ then $ab \bmod N \in \mathbb{Z}^{\times}_N$
  - Since a and b don't have any common factor with N, ab won't have any common factor with N
  - Taking it mod N won't change that
- (Divide) If $a,b \in \mathbb{Z}^{\times}_N$ then there is a unique $x \in \mathbb{Z}^{\times}_N$ such that $ax\equiv b \pmod N$
  - By the usual Euclid's algorithm we can write 1=sa+tN for some s, t with 0<s<N. (a^{-1}=s mod N)
  - Therefore $s \in \mathbb{Z}^{\times}_N$ and so $x=sb \bmod N \in \mathbb{Z}^{\times}_N$

For N=pq, how many elements in $\mathbb{Z}^{\times}_N$ = { a : 0<a<N and gcd(a,N)=1}?

- N=pq and both p and q are prime numbers
  - Only elements between 0 and N-1 not in $\mathbb{Z}^{\times}_N$ are divisible by p or by q
    - There are q different multiples of p
    - There are p different multiples of q
    - Only 0 is a multiple of both
  - Total is N-p-q+1=pq-p-q+1=(p-1)(q-1)
- Standard notation: We write $\varphi(N)=|\mathbb{Z}^{\times}_N|$

Euler's Theorem and RSA

Theorem: For every $a \in \mathbb{Z}^{\times}_N$, $a^{\varphi(N)} \equiv 1 \pmod N$

More generally, for any $a \in \mathbb{Z}^{\times}_N$ and integer $k \geq 0$,

$$a^k \equiv a^{k \bmod \varphi(N)} \pmod N$$

In RSA we want d such that $a^{de} \equiv a^1 \pmod N$

i.e. find a d such that: $1 = de \bmod \varphi(N)$

equivalently, solve: $ex \equiv 1 \pmod{(p-1)(q-1)}$

Can do this if gcd(e, (p-1)(q-1)) = 1.

Theorem: For every $a \in \mathbb{Z}^{\times}_N$, $a^{\varphi(N)} \equiv 1 \pmod N$

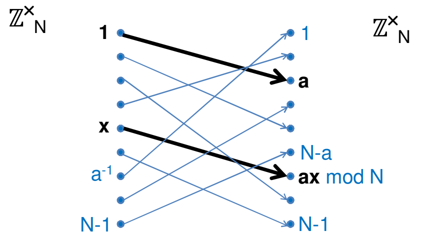
Proof:

Let $a \in \mathbb{Z}^{\times}_N$ and consider function $f_a : \mathbb{Z}^{\times}_N \rightarrow \mathbb{Z}^{\times}_N$ given by $f_a(x) = ax \bmod N$

- Output of f_a is in $\mathbb{Z}^{\times}_N$ by Multiplication property
- f_a is 1-1 by Division property since ab mod N=ac mod N implies $b \equiv c \pmod N$.

We now look at the product of all elements in $\mathbb{Z}^{\times}_N$ modulo N in two different ways

Graph of f_a



Therefore, mod N,
product of all **x** for $\mathbf{x} \in \mathbb{Z}^{\times}_N \equiv$ product of all **ax** for all $\mathbf{x} \in \mathbb{Z}^{\times}_N$

In equations

$$\prod_{x \in \mathbb{Z}^{\times}_N} \mathbf{x} \equiv \prod_{x \in \mathbb{Z}^{\times}_N} \mathbf{ax} \pmod N$$

$$\equiv \mathbf{a}^{\varphi(N)} \prod_{x \in \mathbb{Z}^{\times}_N} \mathbf{x} \pmod N$$

$\prod_{x \in \mathbb{Z}^{\times}_N} \mathbf{x} \bmod N \in \mathbb{Z}^{\times}_N$ by Multiplicative property so we can divide both sides by it to get

$$\mathbf{1} \equiv \mathbf{a}^{\varphi(N)} \pmod N$$

□

Constraints on RSA

- The message has to be in $\mathbb{Z}^{\times}_N$
  - Rule out message 0 and for the rest, you will never see a message divisible by p or q
- The exponent e has to have gcd(e, (p-1)(q-1))=1
  - E.g. p,q will be odd so e can't be even
  - Bob can check this when he chooses e and make sure this doesn't happen